Agnostic Reinforcement Learning with Low-Rank MDPs and Rich Observations

Christoph Dann
Google Research
cdann@cdann.net

Yishay Mansour
Google Research & Tel Aviv University
mansour.yishay@gmail.com

Mehryar Mohri
Google & Courant Institute
mohri@google.com

Ayush Sekhari
Cornell University
as3663@cornell.edu

Karthik Sridharan
Cornell University
ks999@cornell.edu

Abstract

There have been many recent advances on provably efficient Reinforcement Learning (RL) in problems with rich observation spaces. However, all these works share a strong realizability assumption about the optimal value function of the true MDP. Such realizability assumptions are often too strong to hold in practice. In this work, we consider the more realistic setting of agnostic RL with rich observation spaces and a fixed class of policies Π that may not contain any near-optimal policy. We provide an algorithm for this setting whose error is bounded in terms of the rank d of the underlying MDP. Specifically, our algorithm enjoys a sample complexity bound of $\widetilde{O}\left((H^{4d}K^{3d}\log|\Pi|)/\varepsilon^2\right)$ where H is the length of episodes, K is the number of actions and $\varepsilon > 0$ is the desired sub-optimality. We also provide a nearly matching lower bound for this agnostic setting that shows that the exponential dependence on rank is unavoidable, without further assumptions.

1 Introduction

Reinforcement Learning (RL) has achieved several remarkable empirical successes in the last decade, which include playing Atari 2600 video games at superhuman levels [Mnih et al., 2015], AlphaGo or AlphaGo Zero surpassing champions in Go [Silver et al., 2018], AlphaStar's victory over top-ranked professional players in StarCraft [Vinyals et al., 2019], or practical self-driving cars. These applications all correspond to the setting of rich observations, where the state space is very large and where observations may be images, text or audio data. In contrast, most provably efficient RL algorithms are still limited to the classical tabular setting where the state space is small [Kearns and Singh, 2002, Brafman and Tennenholtz, 2002, Azar et al., 2017, Dann et al., 2019] and do not scale to the rich observation setting.

To derive guarantees for large state spaces, much of the existing work in RL theory relies on a *realizability* and a *low-rank* assumption [Krishnamurthy et al., 2016, Jiang et al., 2017, Dann et al., 2018, Du et al., 2019a, Misra et al., 2020, Agarwal et al., 2020a]. Different notions of rank have been adopted in the literature, including that of a low-rank transition matrix [Jin and Luo, 2019], a low Bellman rank [Jiang et al., 2017], Witness rank [Sun et al., 2019], Eluder dimension [Osband and Van Roy, 2014], Bellman-Eluder dimension [Jin et al., 2021], or bilinear classes [Du et al., 2021]. These studies also show that learning without any such structural assumptions requires a sample size that grows exponentially in the time horizon of the MDP [Dann and Brunskill, 2015, Krishnamurthy et al., 2016, Du et al., 2019b]. The choice of the most suitable and most general notion of rank is the topic of much active research in RL theory.

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

In comparison, the realizability assumption has received much less attention. This is the strong premise that the optimal value function belongs to the class of functions considered, which typically does not hold in practice. In many applications, the optimal value function $Q^\star$ is highly complex and we cannot hope to accurately approximate it in the absence of some strong domain knowledge. Can we relax the realizability assumption in RL?

Value-function realizability can be viewed as the analogue of the PAC-realizability assumption in classical statistical learning theory. That assumption rarely holds, which has motivated the development and analysis of numerous algorithms for the agnostic PAC learnability model. Those algorithms provably learn to predict as well as the best predictor in the given function class, independently of whether the Bayes predictor belongs to the class. The counterparts of such results in reinforcement learning are mostly unavailable, which prompts the following question: Can we derive a theory of agnostic reinforcement learning?

Here, we precisely initiate that study. In this agnostic setting, we adopt common structural assumptions, e.g. small rank of the transition matrix, but seek to learn to perform as well as the best policy in the given policy class, independently of how close this class represents the Bellman-optimal policy. Specifically, we study agnostic Reinforcement Learning (RL) with a fixed policy class Π in the episodic MDPs with rich observations. Provably sample-efficient algorithms for agnostic RL would be highly desirable but it is still unknown to what degree learning is possible in this setting. Our work provides new insights about learnability with structural assumptions in the absence of (approximate) realizability in RL.

Agnostic RL without any additional structural assumptions has been considered in the past. By evaluating each policy in the class individually, one can easily obtain a sample complexity upper bound of $O(|\Pi|/\varepsilon^2)$. Kearns et al. [2000] also showed that an upper bound of $(K^H \log|\Pi|)/\varepsilon^2$ is possible, where K is the number of actions and H is the time horizon. However, as discussed in prior work such as [Krishnamurthy et al., 2016], bounds of this form are rather unsatisfactory as one of them admits a linear dependence on the size of the function class, which is prohibitively large, and the other one admits an exponential dependence on the length of the episodes H, which is typically long. Using existing constructions, one can derive a lower bound on the sample complexity of the form $\min\{|\Pi|, K^H\}/\varepsilon^2$ in the rich observation setting. This further justifies our adoption of rank as a natural structural assumption.

Our Contributions: The following highlights our main technical contributions, where d is the rank of the state transition matrix induced by any policy in the class Π, and is assumed to be small.

- We provide a uniform exploration-based algorithm that can find an ε-sub-optimal policy w.r.t. the policy class Π after collecting $O((HK/d)^{4d}\log(d|\Pi|)/\varepsilon^2)$ samples in the MDP. This bound shows that one can achieve a sample complexity that is polynomial in both H and $\log|\Pi|$, while being exponential in rank d only (which we assume is small).

  In addition to the sample complexity bound obtained here, the algorithmic techniques itself might be of independent interest and useful beyond this work. The algorithm is based on showing that for every policy, the expected rewards follows an autoregressive model of degree d. Thus obtaining samples of $O(d)$-length paths for a policy we show that one can extrapolate expected rewards for the entire episode.
- We complement this upper bound with a sample complexity lower bound of $\Omega\big((H/d)^{d/2}/\varepsilon^2\big)$ (when $K = 2$), thereby showing that the $H^{O(d)}$ term in the upper bound is unavoidable. The lower bound also highlights which structures in the policy class induce the $H^{O(d)}$ terms thus shedding light on what structural assumptions could help alleviate the exponential dependence on the rank.
- Finally, we seek to improve upon the H^d term and provide an adaptive algorithm that admits a sample complexity that depends on the eigenspectrum of the transition matrix of the MDP; while in the worst case that bound matches the above one, it provides a significantly better guarantee when the eigenspectrum is more favorable.

However, we view the main benefit of our work to be the initiation of the study of agnostic reinforcement learning and the presentation of an in-depth analysis of a natural structural assumption within that setting. This can form the basis for future research in this domain with alternative and perhaps more favorable rank-type assumptions.

2 Problem Setup

We consider an episodic Markov decision process with episode length $H \in \mathbb{N}$, observation space $\mathcal{X}$ and action space $\mathcal{A} := \{1, \dots, K\}$. For ease of exposition, we assume that the observation space $\mathcal{X}$ is finite (albeit extremely large), but our results can be readily extended to countably infinite and possibly uncountably infinite observation spaces. Each episode is a sequence $((x_1, a_1, r_1), (x_2, a_2, r_2), \dots, (x_H, a_H, r_H)) \in (\mathcal{X} \times \mathcal{A} \times \mathbb{R})^H$, where the initial observation x_0 is drawn from the initial distribution $\mu_0 \in \Delta(\mathcal{X})$, the actions are generated by the learning agent and all the following observations are sampled from the transition kernel $x_{h+1} \sim T(\cdot|x_h, a_h) \in \Delta(\mathcal{X})$ that depends on the previous observation and action. Finally, the rewards r_h are drawn from a sub-Gaussian distribution with mean $r(x_h, a_h)$ where $r\colon \mathcal{X} \times \mathcal{A} \mapsto [0, 1]$. The learning agent does not know the transition kernel T, the initial distribution μ_0, or the reward function r.

In our setting, the agent is given a policy class $\Pi \subseteq \{\mathcal{X} \mapsto \Delta(\mathcal{A})\}$ consisting of policies that map observations to distributions over the actions $\mathcal{A}$. For any policy $\pi \in \Pi$, we denote by $T^\pi \in \mathbb{R}^{\mathcal{X} \times \mathcal{X}}$, the transition matrix induced by π, i.e., for any $x, x' \in \mathcal{X}$,

$$[T^\pi]_{(x',x)} = \mathbb{E}_{a \sim \pi(x)} T(x' \mid x, a).$$

Assumption 1 (Low-rank transition)**.** *There exists* $d \in \mathbb{N}$ *such that* $\operatorname{rank}(T^\pi) \leq d$ *for all* $\pi \in \Pi$.

For the main part of the paper, we assume that the learner knows the value of d, but later extend our results to the case where d is unknown. We define $\lambda^\pi = (\lambda_1^\pi, \dots, \lambda_d^\pi)^\top \in \mathbb{C}^d$ to denote the eigenvalues of the transition matrix T^π with rank at most d. Without any loss of generality, assume that $|\lambda_1^\pi| \geq |\lambda_2^\pi| \geq \dots \geq |\lambda_d^\pi|$.

We denote by $\mathbb{P}^\pi$ the distribution over episodes when following policy π and by $\mathbb{E}^\pi$ its expectation. We call the expected rewards obtained at time h by policy π **expected policy rewards**:

$$R_h^\pi := \mathbb{E}^\pi[r(x_h, a_h)]. \tag{1}$$

The value function of π at time h is given by $V_h^\pi(x) = \mathbb{E}^\pi\left[\sum_{h'=h}^H r(x_{h'}, a_{h'}) \mid x_{h'} = x\right]$. Further, when using V^π without a time index and arguments, we mean the value or expected H-step return:

$$V^\pi := \mathbb{E}[V_0^\pi(x_0)] = \mathbb{E}^\pi\left[\sum_{h=1}^H r(x_h, a_h)\right] = \sum_{h=1}^H R_h^\pi, \tag{2}$$

the value function averaged over initial observations.

Learning objective. The goal of the learner is to return a policy $\widetilde{\pi}$, after interacting with the MDP for n episodes of length H, such that the value of the returned policy is as close as possible to the value of the best policy in Π, that is,

$$V^{\widetilde{\pi}} \geq \max_{\pi \in \Pi} V^\pi - \varepsilon,$$

where the error ε is a small as possible and may depend on n, the policy class Π and the MDP.

3 Related work

We give a brief overview of the most closely related works here, and defer a more detailed discussion to Appendix A.

Recently there has been great interest in designing RL algorithms with general function approximation [Jiang et al., 2017, Dann et al., 2018, Sun et al., 2019, Du et al., 2019a, Wang et al., 2020, Du et al., 2021]. In particular, Jiang et al. [2017] introduced the notion of Bellman rank, a measure of complexity that depends on the underlying environment and the value function class $\mathcal{F}$, and provide statistically efficient algorithms for learning problems for which Bellman rank is bounded. This was later extended to model-based algorithms by Sun et al. [2019]. While these algorithms work across a variety of problem settings, their sample complexity scales with $\log(|\mathcal{F}|)$. Furthermore, these algorithms also require the optimal value function f^* to be realized in $\mathcal{F}$. In our work, we do not

assume that the learner has access to a value function class $\mathcal{F}$. In fact, given a value function class $\mathcal{F}$, we can construct the policy class $\Pi_{\mathcal{F}}$ that corresponds to greedy policies induced by the class $\mathcal{F}$. However, given just a policy class Π, one cannot construct a value function class, without additional knowledge of the underlying dynamics.

Our Assumption 1 implies that for any policy $\pi \in \Pi$, the transition dynamics exhibits a low-rank decomposition with dimension d, that is $T^{\pi}(x'|x) = \langle \phi^{\pi}(x), \psi^{\pi}(x') \rangle$, for some d-dimensional feature maps $\phi^{\pi}, \psi^{\pi} \colon \mathcal{X} \mapsto \mathbb{R}^d$. Low rank MDPs and linear transition models have recently gained a lot of attention in the RL literature [Yang and Wang, 2020, Jin et al., 2020, Modi et al., 2020, Wang et al., 2021a]. The works most closely related to our setup are those of Jin et al. [2020] and Yang and Wang [2020], who give algorithms to find an optimal policy in low rank MDPs with known feature maps ϕ. Similarly, the other algorithms also assume that the learner either observes the feature $\phi(x)$, or the feature $\psi(x)$. Agarwal et al. [2020a] and Modi et al. [2021] learn under weaker assumptions and only assume that the learner has access to a function class that realizes ϕ. However, in our setup, the learner neither observes the features ϕ^{π}, ψ^{π} nor has access to a realizable function class for them, and thus these methods are not applicable.

Several of the works mentioned above recognize the issue of a strict realizability assumption and provide results only when the function class contains a good approximation to the optimal value function of model. However, the goal in our agnostic setting is more ambitious. We would like to find a policy that can compete with the best policy in the given class Π, independent of how close the best return in the class $\max_{\pi \in \Pi} V^{\pi}$ is to the return of the optimal policy $V^{\pi^\star}$ for that MDP.

There have also been several approaches for provably efficient RL with non-parametric function classes [Yang et al., 2020, Long et al., 2021, Shah et al., 2020]. However, these approaches still aim to learn the optimal value function and their regret necessarily scales with the complexity of the optimal value function in the RKHS which can be very high. Instead, in our agnostic setting we would like to be able to quickly identify the best policy from the given policy class with low complexity containing a good but not necessarily optimal policy. Finally, there are a few prior works in agnostic RL that directly compete against a policy class [Abbasi-Yadkori et al., 2013, Azar et al., 2013]. However, they either make strong assumptions on the feedback model, e.g. Abbasi-Yadkori et al. [2013] assumes that the the agent fully observed the current transition kernel and reward instead of just a sample from it, or the provided bound [Azar et al., 2013] scales linearly with the size of the policy class, instead of logarithmically.

4 Upper bound

In this section, we describe our main algorithm for finding a policy that is close to the best-in-class in Π. This algorithm presented in Algorithm 1, is an instance of policy search with uniform exploration. Specifically, we first collect a dataset $\mathcal{D}$ of n episodes by picking actions uniformly at random and subsequently use those episodes to estimate the value of each policy in Π. The algorithm then simply returns the policy $\widetilde{\pi}$ with the highest estimated value.

Our main technical innovation is a new estimation procedure for policy values in Algorithm 2 that leverages the low-rank structure of the transition matrix. A straightforward way to estimate the policy value is to take the sum of the rewards on average across all episodes where all actions are consistent with the policy [Kearns et al., 2000]. Unfortunately, this rejection sampling approach yields an error of $\Omega(\sqrt{K^H})$. Instead, our procedure only estimates the expected policy rewards for the first $3d$ steps. Specifically, when invoked with a given policy π, ValEstimate estimates the expected rewards for that policy by considering the subset of trajectories in $\mathcal{D}$ where π agrees with the chosen action till the first 3d steps, and by averaging the observed rewards in those trajectories. ValEstimate then predicts the future expected rewards for that policy by extrapolating these $3d$ estimated expected rewards. The prediction is computed by recognizing that the expected rewards for any policy π satisfy an autoregressive relation of order d as shown in Lemma 1.

In order to find the coefficients of this autoregression, ValEstimate computes $\widehat{\lambda} \in \mathbb{C}^d$ by solving the optimization problem (4), where the coefficient $\alpha_k(\lambda)$ are the sum of degree k monomials:

$$\alpha_k(\lambda) = \sum_{x \in \{0,1\}^d \text{ s.t. } \|x\|_1 = k} \lambda_1^{x_1} \lambda_2^{x_2} \dots \lambda_d^{x_d}. \tag{3}$$

After estimating $\widehat{\lambda}$, ValEstimate then predicts the expected rewards for all future time steps for the policy π by unfolding the autoregression model whose coefficients are given by $\alpha_k(\widehat{\lambda})$. The estimate for the value of the given policy π, denoted by $\widetilde{V}^\pi$, is then computed as the sum of the predicted expected rewards for H steps.

Finally, Algorithm 1 returns the policy $\widetilde{\pi}$ whose estimated value function is highest amongst all the policies in Π. The following theorem characterizes the performance guarantee for the policy $\widetilde{\pi}$ returned by our algorithm.

Algorithm 1 Policy search algorithm

Input: horizon H, rank d, number of episodes n, finite policy class Π

1: Collect the dataset $\mathcal{D} = \{(x_h^t, a_h^t, r_h^t)_{h=1}^H\}_{t=1}^n$ of n trajectories by drawing actions from Uniform($\mathcal{A}$).
2: **for** policy $\pi \in \Pi$ **do**
3: Estimate $\widetilde{V}^\pi$ by calling ValEstimate$(H, d, \mathcal{D}, \pi)$.
4: **Return:** policy $\widetilde{\pi}$ with best estimated value, i.e. $\widetilde{\pi} \in \operatorname{argmax}_{\pi\in\Pi} \widetilde{V}^\pi$.

Algorithm 2 Value estimation by autoregressive extrapolation

1: **function** VALESTIMATE$(H, d, \mathcal{D}, \pi)$:
2: **for** time step $h = 1, \ldots, 3d$ **do**
3: Estimate expected rewards by importance sampling

$$\widehat{R}_h = \frac{1}{n}\sum_{t=1}^{n} r_h^t \prod_{h'\le h} \left(K\mathbb{1}\{\pi(x_{h'}^t) = a_{h'}^t\}\right)$$

4: Estimate eigenvalues of the autoregression by solving the optimization problem:

$$(\widehat{\lambda}, \widehat{\Delta}) \leftarrow \operatorname*{argmin}_{\lambda\in\mathsf{C}^d, \Delta\in\mathsf{R}} \Delta \qquad \text{s.t. } |\lambda_k| \le 1 \quad \text{for } k = 1, \ldots, d \tag{4}$$

$$\text{and} \quad \left|\sum_{k=1}^{d} (-1)^{k+1}\alpha_k(\lambda)\widehat{R}_{h-k} - \widehat{R}_h\right| \le \Delta \qquad \text{for } h = d+1, \ldots, 3d$$

5: Predict $\widetilde{R}_h$ as:

$$\widetilde{R}_h = \begin{cases} \widehat{R}_h & \text{for } 1 \le h \le d \\ \sum_{k=1}^{d} (-1)^{k+1}\alpha_k(\widehat{\lambda})\widetilde{R}_{h-k} & \text{for } d+1 \le h \le H \end{cases}. \tag{5}$$

6: **return:** Estimate of the value $\widetilde{V} = \sum_{h=1}^{H} \widetilde{R}_h$.

Theorem 1 (Main Theorem). *For a given $\delta \in (0,1)$, d-rank MDP, horizon $H \ge d$ and a finite policy class Π, after collecting n episodes, Algorithm 1 returns a policy $\widetilde{\pi}$ that with probability at least $1-\delta$ admits the following guarantee:*

$$V^{\widetilde{\pi}} \ge \max_{\pi\in\Pi} V^\pi - O\Big(d^3 \cdot \Big(\frac{H}{d}\Big)^{2d} \sqrt{\frac{K^{3d}\log(6\Pi d/\delta)}{n}}\Big).$$

Theorem 1 implies that Algorithm 1 can find an ε-optimal policy with probability $1-\delta$ as long as the number of samples n satisfies

$$n = \Omega\Big(\Big(\frac{H}{d}\Big)^{4d} \frac{K^{3d}\log(6d|\Pi|/\delta)}{\varepsilon^2}\Big).$$

The key idea used in ValEstimate, is that for any policy π for which $\operatorname{rank}(T^\pi) \le d$, the expected rewards satisfy an auto-regression of order d. The following lemma formalize this idea.

Lemma 1 (Autoregression on expected rewards). *Let π be any policy for which the transition matrix T^π has rank at most d. Then, for any time step $h \ge d+1$, the expected reward for policy π at time*

step h, denoted by R_h^π, satisfies the auto-regression

$$R_h^\pi = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda^\pi) R_{h-k}^\pi, \tag{6}$$

where $\lambda^\pi \in \mathbb{C}^d$ denotes the set of eigenvalues of the matrix T^π, and $\alpha_k(\lambda^\pi)$ is as defined in (3).

We defer the proof of Lemma 1 to Appendix C.1. The proof uses the fact that for any policy $\pi \in \Pi$, the distribution over observations at time step h is given by $\mu_h^\pi = (T^\pi)^h \mu_0$, where μ_0 denotes the distribution over the observation space at initialization. If $\mathrm{rank}(T^\pi) \leq d$, an application of the Cayley-Hamilton theorem implies that we can write $(T^\pi)^h$ as a linear combination of $((T^\pi)^{h-1}, \ldots, (T^\pi)^{h-d})$. This implies that μ_h^π, and thus the expected rewards R_h^π, satisfy an auto-regression of order d. While the expected rewards R_h^π satisfy an auto-regression for every policy π, note that we cannot hope for a similar relation between the instantaneous rewards $r_h(s_h^\pi, \pi(s_h))$ observed when taking actions according to π.

The following result shows that we can simultaneously estimate the expected rewards for the first $3d$ steps for every policy $\pi \in \Pi$. Let $\mathcal{D}$ be a dataset of n episodes in the MDP collected by drawing actions uniformly at random from $\mathcal{A}$. Then, for any policy π, there are approximately n/K^{3d} episodes in $\mathcal{D}$ where the actions taken during the first $3d$ time steps matches the predictions of π on those observations. We compute $\widehat{R}_h^\pi$ as the empirical average of the hth step reward in the corresponding n/K^{3d} episodes that match with π for the first $3d$ steps.

Lemma 2 (Importance sampling)**.** *For any $\delta \in (0,1)$, with probability at least $1-\delta$, for any policy $\pi \in \Pi$ and time step $h \in [3d]$, the estimates $\widehat{R}_h^\pi$ computed using importance sampling satisfy the error bound*

$$|\widehat{R}_h^\pi - R_h^\pi| \leq \sqrt{\frac{2K^{3d}\log(6d|\Pi|/\delta)}{n}} + \frac{2K^{3d}\log(6d|\Pi|/\delta)}{n}.$$

For a given policy π, if we had access to the expected rewards $\{R_1^\pi, \ldots, R_d^\pi\}$, we could have solved for the coefficients $\alpha_k(\lambda)$ exactly. However, we only have access to the empirical estimates $\{\widehat{R}_1, \ldots, \widehat{R}_d\}$ of the expected rewards, and thus we compute the coefficients $\alpha_k(\widehat{\lambda})$ by solving the optimization problem in (4). We predict the future expected rewards by extrapolating using $\alpha_k(\widehat{\lambda})$. The following lemma bounds the error propagated due to this mismatch in our estimation.

Lemma 3 (Error propagation bound)**.** *Let $\lambda, \widehat{\lambda} \in \mathbb{C}^d$ be such that $\max\{|\lambda_1|, |\widehat{\lambda}_1|\} \leq 1$. Further, with the initial values $R_1, \ldots, R_d$ and $\widetilde{R}_1, \ldots, \widetilde{R}_d$, let the sequence $\{R_h\}$ and $\{\widetilde{R}_h\}$ be given by*

$$R_h = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) R_{h-k} \qquad \text{and} \qquad \widetilde{R}_h = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}) \widetilde{R}_{h-k},$$

where the coefficients $\alpha_k(\lambda)$ and $\alpha_k(\widehat{\lambda})$ are define as in (3). Then, for all $h \geq 3d+1$,

$$|\widetilde{R}_h - R_h| \leq 2d \cdot \left(\frac{16eh}{d}\right)^{2d} \cdot \max_{h' \leq 3d} |R_{h'} - \widetilde{R}_{h'}|.$$

We defer the proof of Lemma 3 to Appendix C.3. The proof of Theorem 1 follows from combining the above three technical results. Lemma 1 suggests that for any policy $\pi \in \Pi$ for which $\mathrm{rank}(T^\pi) \leq d$, the expected per step rewards satisfy an auto-regression of order at most d. The error propagation bound in Lemma 3 and the bound on the estimation of the expected rewards for the first $3d$ steps given in Lemma 3 implies that, for every policy $\pi \in \Pi$, the estimated value $\widetilde{V}^\pi$ is close to the true value V^π. Specifically, the estimation error in the value of every policy in π is bounded by $\widetilde{O}((H/d)^{2d}\sqrt{K^{3d}\log(|\Pi|)/n})$. Thus, when $n = \widetilde{O}((H/d)^{4d}K^{3d}/\varepsilon^2)$, we have that $|\widetilde{V}^\pi - V^\pi| \leq \varepsilon$ for every policy $\pi \in \Pi$ simultaneously. This implies that the returned policy , that maximize the estimated value $\widetilde{V}^\pi$, is 2ε sub-optimal w.r.t. the best policy in Π. We defer full details of the proof of Theorem 1 to Appendix C.5.

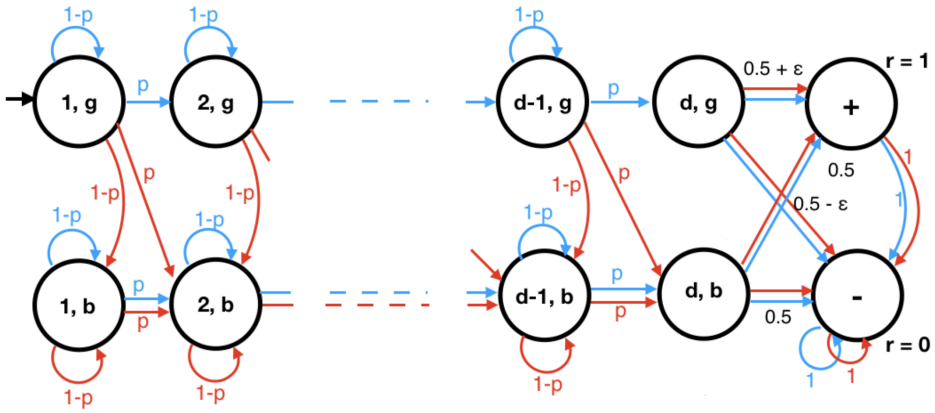

Figure 1: Latent state construction: contextual combination lock. As long as the agent follows actions of π^*(blue arrows), the agent remains in good states (i, g) and receives a Bernoulli$(1/2 + \varepsilon)$ reward but otherwise transits to bad states (i, b) and receives a Bernoulli$(1/2)$ reward.

5 Lower Bound

After presenting an algorithm with sample-complexity bound of $\widetilde{O}((H/d)^{2d}K^{3d}/\varepsilon^2)$, we now show through a lower-bound that the dependency on H and d cannot be improved significantly:

Theorem 2 (Lower bound). *Let $\varepsilon \in (0, 1/26)$, $\delta \in (0, 1/2)$, $d \geq 4$, $K = 2$ and $H \geq 219d$. There exists a policy class of size $(H/d)^d$ and a family of MDPs with rank at most $\Theta(d)$, finite observation space, horizon H and two actions such that the optimal policy for each MDP in the family is contained in the policy class and the following holds: Any algorithm that returns an ε-optimal policy, with probability at least $1 - \delta$, for every MDP in this family has to collect at least*

$$\Omega\Big(\frac{1}{H\varepsilon^2}\Big(\frac{H}{41d}\Big)^{d/2}\log\Big(\frac{1}{2\delta}\Big)\Big)\,.$$

episodes in expectation in some MDP in this family.

The above lower bound shows that an exponential dependency on d in the form of $(H/d)^d$ is unavoidable, even when a realizable policy class with $\pi^* \in \Pi$ and moderate size $\log|\Pi| = d\log(H/d)$ is given to the learner. We now provide a brief description of the problem class used in the proof of our lower bound but defer details of our construction and the proof to Appendix E.

The Markov decision processes in the proof of our lower bound bear some similarity to the so-called *combination lock* constructions used in prior works [Krishnamurthy et al., 2016, Du et al., 2019b], where the algorithm only receives positive feedback after playing a certain sequence of actions. Modelling a combination lock typically requires K^H states in MDPs and $\Theta(H)$ latent states in POMDP. In contrast, our contextual version of a combination lock uses a low-rank MDP with very large observation space but where the transition dynamics are governed by $\Theta(d) \ll H$ hidden states (and thus the rank is $\Theta(d)$). The latent state structure is shown in Figure 1. The agent starts at the top left latent state and always progresses with probability $p = d/H$ to the right. As long as it chooses good actions (blue edges), it progresses in the top chain where it will eventually reach state (d, g) with constant probability and receive a reward of 1 with probability $1/2 + \varepsilon$. If at any time before reaching state (d, g), it chooses a bad action (red edges), then it moves to the lower chain where it eventually has a $1/2$ chance of receiving a reward of 1.

If the latent states $s \in \mathcal{S}$ were directly observable, an ε-optimal policy could be learned with $O(dH/\varepsilon^2)$ samples. However, in the latent state s, the agent only receives an observation drawn uniformly from a large set $\mathcal{X}_s$. The sets $\{\mathcal{X}_s\}_{s\in\mathcal{S}}$ form a partition of the entire observation space $\mathcal{X}$ and there is a mapping $\phi\colon \mathcal{X} \mapsto \mathcal{S}$ that identifies the latent state for each observation. Each MDP $M_{\pi^*,\phi}$ in our problem class is parameterized by the mapping ϕ and a policy $\pi^* \in \Pi$. The class of policies Π can be arbitrary as long as each pair of policies differ on at least a constant fraction of $\mathcal{X}$. In MDP $M_{\pi^*,\phi}$, only the action $\pi^*(x)$ is a good action (blue action) and allows the agent to

stay in the top latent state chain. Thus, finding the $\Theta(\varepsilon)$-best policy in Π for $M_{\pi^*,\phi}$ is equivalent to identifying π^*.

Importantly, our problem class contains MDP $M_{\pi^*,\phi}$ for every possible $\pi^* \in \Pi$ and latent state mapping ϕ. We pick the number of observations large enough so that observations become uninformative and it is virtually impossible for a learner to learn ϕ. Instead it can only hope to learn π^* by identifying the bias ε in the rewards. We can show that this requires number of samples that are not much smaller than collecting $\Theta(1/\varepsilon^2 \ln(1/\delta))$ episodes with each of the $(H/d)^d$ policies in Π.

While the lower bound in Theorem 2 does not have a dependence on $\log(|\Pi|)$. The simple observation that the contextual bandit problem can be seen as an instance of our setup where $d = 1$, implies that some dependence on $\log(|\Pi|)$ is necessary based on standard contextual bandit lower bounds [Lattimore and Szepesvári, 2020]. However, getting a lower bound of the form $\Omega(H^d \log(|\Pi|))$ is an interesting question, which we leave open for future work. Finally, while the provided lower bound is constructed using an MDP with two actions, it can easily be extended to incorporate multiple actions, and when the learner has access to a generative model.

6 Adaptive algorithms

In Section 4, the algorithm introduced benefits from the guarantee provided by Theorem 1, which is near optimal in the worst case as the lower bound construction shows. However, in cases where the transition matrix induced by the policy class all have nicer eigenspectra, one could expect to have an improved sample complexity. Ideally, the algorithm should automatically adapt to more favorable eigenspectra. This is precisely what we describe in this section. We give an adaptive algorithm whose sample complexity improves when the eigenspectrum of transition matrices induced by the policy class admits a more favorable property.

6.1 Adaptivitity to the eigenspectrum

Our adaptive algorithm, presented in Algorithm 3 in Appendix D.3, is a policy search algorithm similar to Algorithm 1 where, instead of invoking the procedure ValEstimate, we compute the value function for every policy π by invoking the procedure AdaValEstimate given in Appendix D.3.

AdaValEstimate follows along the lines of ValEstimate. When invoked for a policy π, it first estimates the expected rewards for the first $3d$ time steps. Then, AdaValEstimate computes the autoregression coefficients $\alpha_k(\widehat{\lambda})$, and uses them to predict the expected rewards for all future time steps by extrapolating. The major difference between ValEstimate and AdaValEstimate is the way the coefficients $\alpha_k(\widehat{\lambda})$ are computed. Specifically, using $\Delta := 2d4^d\sqrt{(8K^{3d}\log(6d|\Pi|)/\delta)/n}$, the procedure AdaValEstimate computes the coefficients $\widehat{\lambda}$ by solving the optimization problem

$$\widehat{\lambda} \leftarrow \operatorname*{argmin}_{\lambda \in \mathsf{C}^d} \prod_{k=2}^{d} \Big(\sum_{h=0}^{H-1} |\lambda_k|^h\Big) \quad \text{s.t. } \lambda_1 = 1, |\lambda_k| \le 1 \qquad \text{for } 2 \le k \le d,$$

$$\text{and } \Big|\sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) \widehat{R}_{h-k} - \widehat{R}_h\Big| \le \Delta \qquad \text{for } d+1 \le h \le 3d.$$

The above modification to the computation of $\widehat{\lambda}$ allows our error propagation bound to adapt to λ, which defines the coefficients of autoregression for the expected rewards in policy π (given in Lemma 1). The propagated error would be small if the coordinates of λ are bounded away from 1. The policy $\widetilde{\pi}$, returned by Algorithm 3, thus enjoys the following adaptive performance guarantee.

Theorem 3 (Adaptive upper bound)**.** *For a given $\delta \in (0,1)$, d-rank MDP, horizon H and a finite policy class Π, after collecting n episodes, Algorithm 3 returns a policy $\widetilde{\pi}$ that with probability at least $1-\delta$ admits the following guarantee:*

$$V^{\widetilde{\pi}} \ge \max_{\pi \in \Pi} V^{\pi} - O\Bigg(dH^2(16e)^{2d} \cdot \max_{\pi' \in \Pi} \prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\lambda_k^{\pi'}|^j\Big)^2 \sqrt{\frac{K^{3d}\log(6\Pi d/\delta)}{n}}\Bigg),$$

Proof of Theorem 3 follows along the lines of the proof of Theorem 1 where we replace the error propagation bound (in Lemma 3) by a similar bound that adapts to the eigenspectrum of the transition matrix T^π. We defer the proof details to appendix Appendix D.3. Note that for any $|\lambda_k| \leq 1$ and thus $\sum_{h=0}^{H-1} |\lambda_k|^h \leq H$. Using this fact in Theorem 3 recovers the result of Theorem 1, albeit upto a multiplicative factor of 2^{2d}. In the following, we provide an example of a low rank MDP problem in which the adaptive bound above could be much better than the worst case upper bound in Theorem 1.

Corollary 1 (Well mixing MDP)**.** *Given $\delta \in (0,1)$, horizon H and a finite policy class Π. Let M be a d-rank MDP such that the second largest eigenvalue of the transition matrix T^π satisfies $|\lambda_2^\pi| \leq 1-\gamma$ for every policy $\pi \in \Pi$. Then, after collecting n episodes, our adaptive algorithm returns a policy $\widetilde{\pi}$ that with probability at least $1-\delta$ admits the following guarantee:*

$$V^{\widetilde{\pi}} \geq \max_{\pi \in \Pi} V^\pi - \widetilde{O}\Big(\Big(\frac{K}{\gamma}\Big)^{2d} \frac{1}{\sqrt{n}}\Big),$$

where the $\widetilde{O}$ hides polynomial factors of $d, H, \log(1/\delta)$ and multiplicative constants.

We next show through a lower bound that the adaptive upper bound in Theorem 3 cannot be improved further. We defer the proof details to Appendix E.3.

Theorem 4 (Adaptive lower bound)**.** *Let $\varepsilon \in (0, 1/16)$, $\delta \in (0, 1/2)$, $d \geq 4$ and $(\lambda_i)_{i\in[d]} \in [0,1]^d$ satisfy*

$$d^{2d} \lesssim \prod_{i=1}^{d} \frac{1}{1-\lambda_i} \lesssim \exp(H) \qquad \text{and} \qquad \sum_{i=1}^{d} \frac{1}{1-\lambda_i} \leq \frac{H}{4\ln(4d)}\,.$$

Then, there is a realizable policy class and a family of MDPs with rank at most $\Theta(d)$, finite observation space, horizon H and two actions such that: For each $i \in [d]$, policy π and MDP M in this class, there is an eigenvalue of the induced transition matrix T_M^π in $[\lambda_i/2, \lambda_i]$. Furthermore, any algorithm that returns, with probability at least $1-\delta$ an ε-optimal policy for any MDP in this family, has to collect at least

$$\Omega\Big(\frac{1}{\varepsilon^2 d^d} \sqrt{\prod_{i=1}^{d} \frac{1}{1-\lambda_i}} \log(1/2\delta)\Big)$$

episodes in expectation in some MDP in this family.

Adaptivity to rank. In Appendix D.4, we also provide an adaptive algorithm that can find the best policy in the class Π without knowing the value of the rank parameter d^*. Our adaptive algorithm, given in Algorithm 5, follows from standard techniques in the model selection literature. For every $d \in [H]$, we compute an optimal policy $\widetilde{\pi}_d$ assuming that the rank $d^* = d$. Then, for each $d \in [H]$, we estimate the value function for the policy $\widetilde{\pi}_d$ by drawing $n/2H$ fresh trajectories using that policy. Finally, we return the policy $\widetilde{\pi}$ from the set $\{\widetilde{\pi}_d\}_{d\in[H]}$ with the highest estimated value. The returned policy $\widetilde{\pi}$ satisfies, with probability at least $1-\delta$,

$$V^{\widetilde{\pi}} \geq \max_{\pi \in \Pi} V^\pi - \widetilde{O}\Big(\Big(\frac{H}{d^*}\Big)^{2d^*} \sqrt{\frac{(8K)^{3d^*} \log(|\Pi|/\delta)}{n}}\Big).$$

We defer full details of the analysis to the Appendix.

7 Conclusion

We presented a new analysis of reinforcement learning with rich observations in the agnostic setting, under the low rank MDP assumption. We gave both a non-adaptive and an adaptive algorithm for learning a quasi-optimal policy in this scenario, which we showed to benefit from guarantees that are only polynomial in the horizon and the number of actions, and only logarithmic in the size of the policy class considered. While our bound is exponential in the MDP rank, we give nearly matching lower bounds proving that that dependency is unavoidable. The agnostic setting is a more realistic setting that has received less attention in the literature. We view this work as initiating the study of this general setting under workable assumptions and believe that many other algorithmic and theoretical aspects of such scenarios need to be studied further.

Acknowledgements

AS was an intern at Google Research, NYC for the duration of the work. KS acknowledges support from NSF CAREER Award 1750575. YM has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 882396), by the Israel Science Foundation (grant number 993/17) and the Yandex Initiative for Machine Learning at Tel Aviv University.

Funding Transparency Statement

Funding in direct support of this work: NSF CAREER Award 1750575, NSF-CCF-1815893, a Google Faculty Fellowship, an NSERC Discovery Grant, and a University of Waterloo startup grant.

References

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575(7782):350–354, 2019.

Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. *Machine learning*, 49(2-3):209–232, 2002.

Ronen I Brafman and Moshe Tennenholtz. R-max-a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3(Oct):213–231, 2002.

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

Christoph Dann, Lihong Li, Wei Wei, and Emma Brunskill. Policy certificates: Towards accountable reinforcement learning. *International Conference on Machine Learning*, 2019.

Akshay Krishnamurthy, Alekh Agarwal, and John Langford. Pac reinforcement learning with rich observations. In *Advances in Neural Information Processing Systems*, pages 1840–1848, 2016.

Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. Contextual decision processes with low Bellman rank are PAC-learnable. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1704–1713. JMLR. org, 2017.

Christoph Dann, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. On oracle-efficient pac rl with rich observations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, 2018.

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

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.

Alekh Agarwal, Sham Kakade, Akshay Krishnamurthy, and Wen Sun. Flambe: Structural complexity and representation learning of low rank mdps. *Advances in Neural Information Processing Systems*, 33, 2020a.

Tiancheng Jin and Haipeng Luo. Learning adversarial markov decision processes with bandit feedback and unknown transition. *arXiv preprint arXiv:1912.01192*, 2019.

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

Ian Osband and Benjamin Van Roy. Model-based reinforcement learning and the eluder dimension. In *Advances in Neural Information Processing Systems*, pages 1466–1474, 2014.

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.

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.

Christoph Dann and Emma Brunskill. Sample complexity of episodic fixed-horizon reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 2818–2826, 2015.

Simon S Du, Sham M Kakade, Ruosong Wang, and Lin F Yang. Is a good representation sufficient for sample efficient reinforcement learning? *arXiv preprint arXiv:1910.03016*, 2019b.

Michael J Kearns, Yishay Mansour, and Andrew Y Ng. Approximate planning in large pomdps via reusable trajectories. In *Advances in Neural Information Processing Systems*, pages 1001–1007, 2000.

Ruosong Wang, Russ R Salakhutdinov, and Lin Yang. Reinforcement learning with general value function approximation: Provably efficient approach via bounded eluder dimension. *Advances in Neural Information Processing Systems*, 33, 2020.

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

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

Aditya Modi, Nan Jiang, Ambuj Tewari, and Satinder Singh. Sample complexity of reinforcement learning using linearly combined model ensembles. In *International Conference on Artificial Intelligence and Statistics*, pages 2010–2020. PMLR, 2020.

Yining Wang, Ruosong Wang, Simon Shaolei Du, and Akshay Krishnamurthy. Optimism in reinforcement learning with generalized linear function approximation. In *International Conference on Learning Representations*, 2021a.

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

Zhuoran Yang, Chi Jin, Zhaoran Wang, Mengdi Wang, and Michael I Jordan. On function approximation in reinforcement learning: Optimism in the face of large state spaces. *arXiv preprint arXiv:2011.04622*, 2020.

Jihao Long, Jiequn Han, et al. An l2 analysis of reinforcement learning in high dimensions with kernel and neural network approximation. *arXiv preprint arXiv:2104.07794*, 2021.

Devavrat Shah, Dogyoon Song, Zhi Xu, and Yuzhe Yang. Sample efficient reinforcement learning via low-rank matrix estimation. *arXiv preprint arXiv:2006.06135*, 2020.

Yasin Abbasi-Yadkori, Peter L Bartlett, and Csaba Szepesvári. Online learning in markov decision processes with adversarially chosen transition probability distributions. *arXiv preprint arXiv:1303.3055*, 2013.

Mohammad Gheshlaghi Azar, Alessandro Lazaric, and Emma Brunskill. Regret bounds for reinforcement learning with policy advice. In *Joint European Conference on Machine Learning and*

Knowledge Discovery in Databases, pages 97–112. Springer, 2013.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(Apr):1563–1600, 2010.

Tor Lattimore and Marcus Hutter. Pac bounds for discounted mdps. In *International Conference on Algorithmic Learning Theory*, pages 320–334. Springer, 2012.

Sham M Kakade. A natural policy gradient. *Advances in neural information processing systems*, 14, 2001.

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

Sergey Levine and Vladlen Koltun. Guided policy search. In *International conference on machine learning*, pages 1–9. PMLR, 2013.

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

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

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *arXiv preprint arXiv:1908.00261*, 2019.

Yasin Abbasi-Yadkori, Peter Bartlett, Kush Bhatia, Nevena Lazic, Csaba Szepesvari, and Gellért Weisz. Politex: Regret bounds for policy iteration using expert prediction. In *International Conference on Machine Learning*, pages 3692–3702. PMLR, 2019.

Jalaj Bhandari and Daniel Russo. Global optimality guarantees for policy gradient methods. *arXiv preprint arXiv:1906.01786*, 2019.

Yanli Liu, Kaiqing Zhang, Tamer Basar, and Wotao Yin. An improved analysis of (variance-reduced) policy gradient and natural policy gradient methods. *Advances in Neural Information Processing Systems*, 33, 2020.

Alekh Agarwal, Mikael Henaff, Sham Kakade, and Wen Sun. Pc-pg: Policy cover directed exploration for provable policy gradient learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 13399–13412. Curran Associates, Inc., 2020b.

Qiang Liu, Lihong Li, Ziyang Tang, and Dengyong Zhou. Breaking the curse of horizon: Infinite-horizon off-policy estimation. *arXiv preprint arXiv:1810.12429*, 2018.

Ofir Nachum, Yinlam Chow, Bo Dai, and Lihong Li. Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections. *arXiv preprint arXiv:1906.04733*, 2019.

Ruosong Wang, Dean Foster, and Sham M. Kakade. What are the statistical limits of offline {rl} with linear function approximation? In *International Conference on Learning Representations*, 2021b.

Andrea Zanette. Exponential lower bounds for batch reinforcement learning: Batch rl can be exponentially harder than online rl. *arXiv preprint arXiv:2012.08005*, 2020.

Gellert Weisz, Philip Amortila, and Csaba Szepesvári. Exponential lower bounds for planning in mdps with linearly-realizable optimal action-value functions. In *Algorithmic Learning Theory*, pages 1237–1264. PMLR, 2021.

J Segercrantz. Improving the cayley-hamilton equation for low-rank transformations. *The American mathematical monthly*, 99(1):42–44, 1992.

DJ Hartfiel. Tracking in matrix systems. *Linear algebra and its applications*, 165:233–250, 1992.

Darald J Hartfiel. Dense sets of diagonalizable matrices. *Proceedings of the American Mathematical Society*, 123(6):1669–1672, 1995.

Rajendra Bhatia. *Matrix analysis*, volume 169. Springer Science & Business Media, 2013.

Pascal Massart. *Concentration inequalities and model selection*. Springer, 2007.

Aurélien Garivier, Pierre Ménard, and Gilles Stoltz. Explore first, exploit next: The true shape of regret in bandit problems. *Mathematics of Operations Research*, 44(2):377–399, 2019.

Omar Darwiche Domingues, Pierre Ménard, Emilie Kaufmann, and Michal Valko. Episodic reinforcement learning in finite mdps: Minimax lower bounds revisited. In *Algorithmic Learning Theory*, pages 578–598. PMLR, 2021.

Contents of Appendix

A Detailed comparison to prior work

Provably sample-efficient learning algorithms have been well studied in the classical tabular RL literature [Kearns and Singh, 2002, Brafman and Tennenholtz, 2002]. However, the number of samples required by these algorithms to find the optimal policy π^* scales with the size of the state space $|\mathcal{X}|$ [Jaksch et al., 2010, Lattimore and Hutter, 2012], and thus these methods fail to scale to the rich observation settings where $|\mathcal{X}|$ could be astronomically large. There have been significant recent advances in developing efficient algorithms for such rich observation settings, albeit under additional assumptions. The two main styles of assumptions considered in the literature to make learning tractable are: (a) the learner has access to a value function class $\mathcal{F}$ that realizes the optimal value function f^* for the underlying MDP, and (b) the underlying transition dynamics admits additional structure such as low rank or linear decomposition, etc. We note that the goal of these works is to find the optimal policy for the underlying MDP. In comparison, in our work, we assume access to a policy class Π and our goal is to find a policy $\widetilde{\pi}$ that could compete with the best policy in the class Π. In the following, we compare our setup with the assumptions made in the prior work.

RL with general value function approximation. Recently there has been great interest in designing RL algorithms with general function approximation [Jiang et al., 2017, Dann et al., 2018, Sun et al., 2019, Du et al., 2019a, Wang et al., 2020]. In particular, Jiang et al. [2017] introduced the notion of Bellman rank, a measure of complexity that depends on the underlying environment and the value function class $\mathcal{F}$, and provide statistically efficient algorithms for learning problem for which Bellman rank is bounded. This was later extended to model-based algorithms by Sun et al. [2019]. While these algorithms work across a variety of problem settings, their sample complexity scales with $\log(|\mathcal{F}|)$. Furthermore, these algorithms also require the optimal value function f^* to be realized in $\mathcal{F}$. In our work, we do not assume that the learner has access to a value function class $\mathcal{F}$. In fact, given a value function class $\mathcal{F}$, we can construct the policy class $\Pi_{\mathcal{F}}$ that corresponds to greedy policies induced by the class $\mathcal{F}$. However, given just a policy class Π, one can not construct a value function class, without additional knowledge of the underlying dynamics.

Example 1. *Let $\mathcal{X} = \{0, 1, \ldots, N\}$, $\mathcal{A} = \{0, 1\}$, $\Pi = \{\pi_0, \pi_1\}$ and $H = 2$. For every action $a \in \mathcal{A}$, we define the reward $r(x, a) = 1$ when x is even, and $r(x, a) = 0$ when x is odd. Further, we assume that the transition dynamics T is parameterized by a vector $p \in \{0, 1\}^N$ such that for any state x, if $p(x) = 1$, then the next state x' is sampled uniformly at random from the set of even numbers in $\mathcal{X}$, independent of the chosen action. When $p(x) = 0$, we sample an odd number uniformly at random for x'. Thus, in order to learn the optimal value function, the leaner needs to recover the value of the vector p on at least $O(N)$ states. From standard packing arguments, we get that in N dimensions there are at least $2^{O(N)}$ vectors that are $O(N)$ apart. Thus, any appropriate value function class $\mathcal{F}$ that contains p must have size at least $2^{O(N)}$.*

Linear MDP assumption. Our Assumption 1 implies that for any policy $\pi \in \Pi$, the transition dynamics exhibits a low-rank decomposition with dimension d, that is $T^\pi(x'|x) = \langle \phi^\pi(x), \psi^\pi(x') \rangle$,, for some d-dimensional feature maps $\phi^\pi, \psi^\pi \colon \mathcal{X} \mapsto \mathbb{R}^d$. Low rank MDPs and linear transition models have recently gained a lot of attention in the RL literature [Yang and Wang, 2020, Jin et al., 2020, Modi et al., 2020, Wang et al., 2021a]. The works most closely related to our setup are those of Jin et al. [2020] and Yang and Wang [2020], who give algorithms to find optimal policy in low rank MDPs with known feature maps ϕ. Similarly, the other algorithms also assume that the learner either observes the feature $\phi(x)$, or the feature $\psi(x)$. However, in our setup, the learner neither observes the features ϕ^π nor the features ψ^π, thus restricting application of these algorithms to our setting.

A new line of work, initiated by Agarwal et al. [2020a], focuses on the representation learning question in the above setting. They assume that the feature functions ϕ and ψ, although not known to the learner, are realized in the given classes Φ and Ψ respectively. In order to find the optimal policy, their algorithm first identifies the underlying feature functions ϕ^* and ψ^*, and thus, their sample complexity guarantees scale with $\log(|\Phi||\Psi|)$. Later, Modi et al. [2021] show that a similar approach also works when the learner has only access to a Φ but not Ψ. In comparison, we do not assume knowledge of either classes Φ or Ψ, and instead work with a policy class Π. In fact, the following simple illustrative example shows that the feature function Φ could be arbitrarily complex even when $|\Pi|$ is small, and thus we can not hope to learn the feature function from samples.

Example 2. *Let $\mathcal{X} = [N]$, $\mathcal{A} = \{0, 1\}$. We define the feature function $\psi(x) \in \mathbb{R}^2$ such that $(1/2N, 0)^\top$ if x is even and $(0, 1/2N)^\top$ if x is odd. Further, for $\lambda \geq 0$, define the feature function $\phi_\lambda(x) \in \mathbb{R}^2$ such that $\phi_\lambda(x) = (1, 0)^\top$ if $\sin(x/\lambda) \geq 0$, and $\phi_\lambda(x) = (0, 1)^\top$ otherwise. In this MDP, the next state x' is either sampled uniformly at random from the set of even numbers in $\mathcal{X}$ or sampled uniformly at random from the set of odd numbers in $\mathcal{X}$, depending on the value of* $\sin(x/\lambda)$.

Note that the mapping $x \mapsto \sin(x/\lambda)$ could be arbitrarily complex when λ is small. In fact, the function class $\Phi = \{\lambda \mid x \mapsto \sin(x/\lambda)\}$ has infinite VC dimension. Thus, one cannot hope to learn the feature function ϕ_λ from samples.

It is worth noting that in the above example, FLAMBE [Agarwal et al., 2020a], MOFFLE [Modi et al., 2021], or in fact any other approach that attempts to recover the feature function ϕ, as mentioned above will not succeed. Furthermore, when $|\Pi|$ is large and the length of the episode H is large, the previously known agnostic upper bounds of $\frac{|\Pi|}{\varepsilon^2}$ or $\frac{2^H \log(|\Pi|)}{\varepsilon^2}$ are also prohibitively large. However, in the above example, our algorithm enjoys a sample complexity bound of $\frac{H^4 \log(|\Pi|)}{\varepsilon^2}$.

Finally, note that in our setup, the decomposition of the induced transition kernel (into ϕ^π and ψ^π) may be different for each policy π in the class Π. Furthermore, there may be policies outside of Π that do not even exhibit such a low-rank decomposition. Thus, although our low rank assumption is similar to those in linear or low-rank MDPs [Agarwal et al., 2020a], our model is more general.

Comparison to Block MDP model. Krishnamurthy et al. [2016] introduced the block MDP model, where a small number of latent states $\mathcal{S}$ govern the transition dynamics, and the observations $x \in \mathcal{X}$ are generated depending on the current latent state s. In this model, there is a decoding function g^* that maps observations x back to the latent state s that generates x. Du et al. [2019a], Misra et al. [2020] assume that the learner is given a realizable class of decoding functions $\mathcal{G}$ and show that the true mapping $g^* \in \mathcal{G}$ can be learnt efficiently, both computationally and statistically, which can then be used to find the optimal policy. However, note that the transition matrix in a Block MDPs with S latent states has rank at most $|S|$, and thus their model is captured by our Assumption 1. However, in our setup, we do not assume that the leaner has access to the class $\mathcal{G}$. In fact, Example 2 above shows that the latent state map g^* (the mapping $\phi_\lambda(x)$ in that case) could be arbitrarily complex even when Π is small, and thus we can not hope to learn g^* from samples.

Policy gradient methods. Model free direct policy search algorithms that directly maximize the value function have shown tremendous empirical success [Kakade, 2001, Kakade and Langford, 2002, Levine and Koltun, 2013, Schulman et al., 2015, 2017], and recently, have been analysed from a theoretical perspective [Agarwal et al., 2019, Abbasi-Yadkori et al., 2019, Bhandari and Russo, 2019, Liu et al., 2020, Agarwal et al., 2020b]. While these methods operate directly on a policy class Π, as we do in our work, they require additional modelling assumptions in order to succeed; foremost being that the policy class Π exhibits a differentiable paraeterization. Further assumptions include that the policy class Π contains the optimal policy π^*, the policy class Π has a good coverage over the state space [Agarwal et al., 2019], and that the underling MDP has a linear factorization with known feature maps [Agarwal et al., 2020b]. We do not require these assumptions.

DICE/DualDICE algorithms. Recent works of Liu et al. [2018] and Nachum et al. [2019] provide estimators that do not suffer the curse of horizon, i.e. the factor of A^H, in off-policy estimation of expected policy rewards by applying importance sampling on average visitation distributions of single steps of state-action pairs, instead of the much higher dimensional distribution of the whole trajectories. However, their estimator requires access to a function class $\mathcal{H}$ that contains the importance weights of the average visitation distribution. We do not require access to such a class $\mathcal{H}$ in our estimator of expected policy rewards.

POMDP with reactive policies. We will show in the following that our theory and algorithm applies to partially observable Markov decision processes (POMDPs), as long as policies are reactive, that is, only take the current observation into account. Although existing works such as [Jiang et al., 2017] show polynomial sample-complexity bounds for POMDPs with reactive policy classes, they require the *optimal* policy to be reactive, which is not true in POMDPs in general. In contrast, we can handle the important scenario where reactive policies can achieve good but not necessarily close to optimal performance and we are interested in finding the best such policy.

A POMDP consists of a MDP with finite state space $\mathcal{S}$, action space $\mathcal{A}$ and horizon H where observed rewards at each step are drawn from a distribution with mean $r(s_h, a_h)$ that depends on the current state s_h and action a_h. Similarly, the next state is drawn fron a transition kernel $P(s_{h+1}|s_h, a_h)$. However, in a POMDP, the current state is not observable and the agent instead receives an observation $x_h \in \mathcal{X}$. We consider the formulation where the observation is drawn from a distribution $O(x_h|s_h)$ that depends on the current latent state s_h. Unlike in, e.g., Block MDP models, x_h does not need to be sufficient to decode s_h and this model does not need to be an MDP over the observation space $\mathcal{X}$. As a consequence, the optimal actions do in general depend on *all* previous observations. Nonetheless, reactive policies which are of the form $\mathcal{X} \to \mathcal{A}$ and only take the current observation into account, often achieve good performance and are of particular interest in practice due to their simplicity.

Since a POMDP may not be a MDP over observations, such models are formally outside of our scope. However, as our technique never explicitly accesses observations except through the policy, we can cast a POMDP problem as follows in our framework. For any policy $\pi \colon \mathcal{X} \to \mathcal{A}$ in our policy class Π we define a stochastic policy π' over latent states as $\pi'(a|s) = \sum_{x \in \mathcal{X}} \mathbb{1}\{\pi(x) = a\} O(x|s)$ and denote the class of these policies by $\Pi' \subseteq \mathcal{S} \to \Delta(\mathcal{A})$. Running our algorithms on a POMDP with policy class Π is equivalent to running them on an MDP with direct access to latent states $\mathcal{S}$ and policy class Π'. Since an MDP with finite state space $\mathcal{S}$ has rank at most $|\mathcal{S}|$, our guarantees apply to POMDPs with a reactive policy class and we can set $d = |\mathcal{S}|$.

Exponential lower bounds for planning and offline RL. Several publications [Wang et al., 2021b, Zanette, 2020, Weisz et al., 2021] recently provide exponential lower bounds for learning the optimal policy with access to a realizable linear Q-function class $\mathcal{F}$ of dimension d in several settings. Most related is Wang et al. [2021b], wo study offline RL where the agent has only access to a dataset of transition samples and show even if the dataset has good coverage of the features of $\mathcal{F}$, a sample complexity that is exponential in d or H is unavoidable. In contrast, we allow the agent to collect samples arbitrarily by interacting with the MDP and although our algorithms first collect a dataset non-adaptively, the uniform action choices ensure good state coverage as opposed to just feature coverage which avoids the existing lower bounds.

B Cayley-Hamilton theorems

The following result holds for any matrix A with rank d.

Lemma 4 (Cayley-Hamilton Theorem for rank d matrices [Segercrantz, 1992])**.** *Let $A \in \mathbb{C}^{N\times N}$ be a matrix with rank at most d, where $d \le N$, and let $\lambda = (\lambda_1, \ldots, \lambda_d) \in \mathbb{C}^d$ denote the set of eigenvalues of A. Then, A satisfies the relation*

$$A^{d+1} = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) A^{d+1-k},$$

where the coefficient $\alpha_k(\lambda)$ are given by the sum of degree k monomials:

$$\alpha_k(\lambda) = \sum_{x\in\{0,1\}^d \text{ s.t. } \|x\|_1 = k} \lambda_1^{x_1} \lambda_2^{x_2} \ldots \lambda_d^{x_d}.$$

The proof of the above follows from the characteristic polynomial for rank d matrices, which allows us to express $d+1$-th power for any matrix A in terms of the lower powers.

We will soon provide an extension of the above result which allows us to express the n-th power of the matrix A in terms of the lower powers. Before doing so, we need to define some additional notation.

B.1 Coefficients $\alpha_{m,k}$

For any $m \ge 0$ and $k \ge 0$, we first define the coefficients $\alpha_{m,k}$.

Definition 1. *For any $k \ge 0$ and $\lambda = (\lambda_1, \ldots, \lambda_d) \in \mathbb{C}^d$, define $\alpha_{m,k}(\lambda)$ to denote the quantity*

$$\alpha_{m,k}(\lambda) := \sum_{y\in\{0,\ldots,m\}^d} \mathbb{1}\Big\{\sum_{j=1}^{d} \mathbb{1}\{y_j > 0\} = k \text{ and } \sum_{j=1}^{d} y_j = m\Big\} \prod_{j=1}^{d} \lambda_j^{y_j}. \tag{7}$$

whenever $m \ge k$ and $\alpha_{m,k} = 0$ when $m \le k$ or $k > d$. Further, for the ease of notation, for any $k \in [d]$, we define $\alpha_k(\lambda)$ to denote the quantity $\alpha_{k,k}(\lambda)$.

The following lemma provides a useful technical relation between the coefficients $\alpha_{m,k}$ defined above.

Lemma 5. *For any $m \ge 0$, $k \in [d]$ and $\lambda \in \mathbb{C}^d$, the quantities $(\alpha_{m,k})_{k\in[d],m\ge 0}$ given in Definition 1 satisfy*

$$\sum_{j=1}^{m\wedge d} \alpha_{m,j}(\lambda) \cdot \alpha_{k,k}(\lambda) = \sum_{j'=k+1}^{(m+k)\wedge d} \binom{j'}{k} \alpha_{m+k,j'}(\lambda).$$

Proof. For the sake of the proof, we will be interpreting $\alpha_{m,j}(\lambda)$ and $\alpha_{k,k}(\lambda)$ as symmetric polynomials with λ as the formal variables. The value of these quantities can be computed by plugging in the value of $\lambda_1, \ldots, \lambda_d$ for λ.

Thus, $\alpha_{m,j}$ denotes a symmetric sum of monomials, where each monomial term has j variables with sum of all the powers in that monomial being m. Similarly, $\alpha_{k,k}$ denotes a symmetric sum of monomials, where each monomial term has k variables each with the power of 1. Subsequently, when we take the product $\alpha_{m,j} \cdot \alpha_{k,k}$, we will get monomial terms, where in each term the sum of all the powers is $m+k$, but the total number of distinct variables can range from $k+1$ to $\min\{j+k, d\}$. Since, the polynomials $\alpha_{m,j}$ and $\alpha_{k,k}$ are symmetric in λ, the resultant polynomial that we will get after taking their product will also be symmetric. Furthermore, each of the monomial terms with j' distinct variables can be generated through $\binom{j'}{k}$ different splits with k variables that go into $\alpha_{k,k}(\lambda)$ and the rest $j'-k$ variables that go into $\alpha_{m,j'-k}(\lambda)$. Hence, the coefficient of $\alpha_{m+k,j'}$ would be exactly $\binom{j'}{k}$. We formalize this in the following:

$$\sum_{j=1}^{m\wedge d} \alpha_{m,j}(\lambda) \cdot \alpha_{k,k}(\lambda)$$

$$
\begin{aligned}
&= \sum_{j=1}^{m\wedge d}\Bigg(\sum_{y\geq 0}\mathbb{1}\Big\{\sum_{i=1}^{d}\mathbb{1}\{y_i>0\}=j \text{ and } \sum_{i=1}^{d} y_i = m\Big\}\prod_{i=1}^{d}\lambda_i^{y_i}\times \\
&\qquad\qquad \sum_{y'\geq 0}\mathbb{1}\Big\{\sum_{i=1}^{d}\mathbb{1}\{y'_i>0\}=k \text{ and } \sum_{i=1}^{d} y'_i = k\Big\}\prod_{i=1}^{d}\lambda_i^{y'_i}\Bigg) \\
&= \sum_{j=1}^{m\wedge d}\sum_{y,y'\geq 0}\mathbb{1}\Big\{\sum_{i=1}^{d}\mathbb{1}\{y_i>0\}=j\wedge\sum_{i=1}^{d}\mathbb{1}\{y'_i>0\}=k\wedge\sum_{i=1}^{d} y_i = m\wedge\sum_{i=1}^{d} y'_i = k\Big\}\prod_{i=1}^{d}\lambda_i^{y_i+y'_i} \\
&= \sum_{j'=k+1}^{(m+k)\wedge d}\binom{j'}{k}\sum_{y,y'\geq 0}\mathbb{1}\Big\{\sum_{i=1}^{d}\mathbb{1}\{y_i+y'_i>0\}=j' \text{ and } \sum_{i=1}^{d} y_i+y'_i = m+k\Big\}\prod_{i=1}^{d}\lambda_i^{y_i+y'_i} \\
&= \sum_{j'=k+1}^{(m+k)\wedge d}\binom{j'}{k}\sum_{y''\geq 0}\mathbb{1}\Big\{\sum_{i=1}^{d}\mathbb{1}\{y''_i>0\}=j' \text{ and } \sum_{i=1}^{d} y''_i = m+k\Big\}\prod_{i=1}^{d}\lambda_i^{y''_i} \\
&= \sum_{j'=k+1}^{(m+k)\wedge d}\binom{j'}{k}\cdot\alpha_{m+k,j'},
\end{aligned}
$$

where $y''_i := y_i + y'_i$ and the third equality in the above follows by rearranging the terms while satisfying the constraints inside the indicator. □

We next provide a bound on the value of $\alpha_{m,k}$ as a function of m and k.

Lemma 6. *For any $d \geq 1$, $m \geq 0$, $k \leq \min\{d, m\}$ and $\lambda \in \mathbb{C}^d$, that satisfies $|\lambda_j| \leq 1$ for all $j \in [d]$, the quantities $\alpha_{m,k}(\lambda)$ given in Definition 1 satisfy the bound*

$$
|\alpha_{m,k}(\lambda)| \leq \Big(\frac{4e\max\{m,d\}}{d}\Big)^d.
$$

Furthermore, for $k = m \leq d$, we have that $\alpha_k(\lambda) = \alpha_{k,k}(\lambda) \leq 4^d$.

Proof. Starting from the definition of $\alpha_{m,k}(\lambda)$, we note that

$$
\begin{aligned}
|\alpha_{m,k}(\lambda)| &= \Big|\sum_{y\in\{0,\ldots,m\}^d}\mathbb{1}\Big\{\sum_{j=1}^{d}\mathbb{1}\{y_j>0\}=k \text{ and } \sum_{j=1}^{d} y_j = m\Big\}\prod_{j=1}^{d}\lambda_j^{y_j}\Big| \\
&\leq \sum_{y\in\{0,\ldots,m\}^d}\mathbb{1}\Big\{\sum_{j=1}^{d}\mathbb{1}\{y_j>0\}=k \text{ and } \sum_{j=1}^{d} y_j = m\Big\}\Big|\prod_{j=1}^{d}\lambda_j^{y_j}\Big| \\
&= \sum_{y\in\{0,\ldots,m\}^d}\mathbb{1}\Big\{\sum_{j=1}^{d}\mathbb{1}\{y_j>0\}=k \text{ and } \sum_{j=1}^{d} y_j = m\Big\}\prod_{j=1}^{d}|\lambda_j^{y_j}| \\
&\leq \sum_{y\in\{0,\ldots,m\}^d}\mathbb{1}\Big\{\sum_{j=1}^{d}\mathbb{1}\{y_j>0\}=k \text{ and } \sum_{j=1}^{d} y_j = m\Big\},
\end{aligned}
$$

where the inequality in the second line follows from an application of Triangle inequality. The last line holds because $|\lambda_j| \leq 1$, and thus $|\prod \lambda_j^{y_j}| \leq 1$ for any y. We note that the right hand side in the above expression denotes the number of ways of distributing m balls into d bins such that exactly k of them are non-empty. If $m = k = 1$, we get that $|\alpha_{m,k}(\lambda)| \leq 1$. Otherwise, a simple counting argument implies that

$$
|\alpha_{m,k}(\lambda)| \leq \binom{d}{k}\binom{m-1}{k-1} \leq 2^d\cdot\binom{m-1}{k-1}.
$$

When $m \leq d$ or $k = 1$, we can simply upper bound the above as

$$
|\alpha_{m,k}(\lambda)| \leq 2^d\cdot 2^m \leq 4^d.
$$

Next, when $m > d$ and $k \geq 2$, using the fact that $\binom{N}{n} \leq (eN/n)^n$ for any $0 < n \leq N$, we get that

$$\begin{aligned}|\alpha_{m,k}(\lambda)| &\leq 2^d \cdot \left(\frac{e(m-1)}{(k-1)}\right)^k \\ &\leq 2^d \cdot \left(\frac{2em}{k}\right)^k \\ &\leq 2^d \cdot \left(\frac{2em}{d}\right)^d,\end{aligned}$$

where the inequality in the second line above holds because $(m-1)/(k-1) \leq 2m/k$ for $k \geq 2$, and the inequality in the last line holds because the function $(x/y)^y$ is an increasing function of y when $x \geq ey$.

Considering the above two bounds together implies that:

$$|\alpha_{m,k}(\lambda)| \leq \left(\frac{4e\max\{m,d\}}{d}\right)^d.$$

□

B.2 Coefficients $\beta_{m,k}$

We next define the coefficients $\beta_{m,k}$ which will be useful in our upper bound analysis.

Definition 2. *For any $d \geq 1$, $\lambda \in \mathbb{C}^d$ and $m \geq 0$, define the vector $\beta_m(\lambda) \in \mathbb{C}^d$ using the following recursion:*

(a) $\beta_0(\lambda) := (\alpha_1(\lambda), \ldots, \alpha_d(\lambda))^\top$, and,

(b) For $m \geq 1$, define $\beta_m(\lambda) := (\beta_{m,1}(\lambda), \ldots, \beta_{m,d}(\lambda))^\top$ as

$$\beta_{m,k}(\lambda) = \begin{cases} \beta_{m-1,1}(\lambda) \cdot \alpha_k(\lambda) - \beta_{m-1,k+1}(\lambda) & \text{for } 1 \leq k \leq d-1 \\ \beta_{m-1,d}(\lambda) \cdot \alpha_d(\lambda) & \text{for } k = d \end{cases},$$

where $\alpha_k(\lambda)$ is as defined in (7) , and $\beta_{0,k}$ denotes the k-th coordinate of the vector β_0.

The next technical lemma provides a relation between the β and α values defined above.

Lemma 7. *For any $m \geq 0$ and $k \in [d]$,*

$$\beta_{m,k}(\lambda) = \sum_{j=k}^{(m+k)\wedge d} \binom{j-1}{k-1} \alpha_{m+k,j}(\lambda). \tag{8}$$

Proof. We prove the desired relation via induction over m. For the base case, when $m = 0$, from the definition of $\beta_{0,k}$, we note that

$$\beta_{0,k} = \alpha_{k,k}(\lambda) = \sum_{j=k}^{k} \binom{k-1}{k-1} \alpha_{k,j}(\lambda).$$

Now, we proceed to the induction step. Assume that the relation (8) holds for all $m' < m$. Thus, for any $k \in [d]$, from the definition of $\beta_{m,k}(\lambda)$, we have that

$$\begin{aligned}\beta_{m,k}(\lambda) &= \beta_{m-1,1}(\lambda) \cdot \beta_{0,k}(\lambda) - \beta_{m-1,k+1}(\lambda) \\ &= \left(\sum_{j=1}^{m\wedge d} \alpha_{m,j}(\lambda)\right) \cdot \alpha_{k,k}(\lambda) - \sum_{j=k+1}^{(m+k)\wedge d} \binom{j-1}{k} \cdot \alpha_{m+k,j}(\lambda),\end{aligned}$$

where the equality in the second line follows from using the relation (8) for time step $m-1$. Using the identity in Lemma 5 in the above, we get that

$$\beta_{m,k}(\lambda) = \sum_{j=k}^{(m+k)\wedge d} \binom{j}{k} \cdot \alpha_{m+k,j}(\lambda) - \sum_{j=k+1}^{(m+k)\wedge d} \binom{j-1}{k} \cdot \alpha_{m+k,j}(\lambda)$$

$$= \sum_{j=k}^{(m+k)\wedge d} \binom{j-1}{k-1} \cdot \alpha_{m+k,j}(\lambda),$$

where the last line uses the relation $\binom{j}{k} = \binom{j-1}{k-1} + \binom{j-1}{k}$. This completes the induction step. Thus, proving that the relation (8) holds for all $m \geq 0$ and $k \in [d]$. □

We next provide a bound on the value of the coefficients $\beta_{m,k}$ as a function of m and k.

Lemma 8. *For any $d \geq 1$, $m \geq 0$, $k \leq d$ and $\lambda \in \mathbb{C}^d$, such that $|\lambda_j| \leq 1$ for all $j \in [d]$, the quantities $\beta_{m,k}(\lambda)$ defined in Definition 2 satisfy the bound*

$$|\beta_{m,k}(\lambda)| \leq \Big(\frac{8e \max\{m+k, d\}}{d}\Big)^d.$$

Proof. As a consequence of Lemma 7, we have that for any $m \geq 0$ and $k \in [d]$,

$$\beta_{m,k}(\lambda) = \sum_{j=k}^{(m+k)\wedge d} \binom{j-1}{k-1} \cdot \alpha_{m+k,j}(\lambda).$$

Thus, using Triangle inequality, we have that

$$\begin{aligned} |\beta_{m,k}| &= \Big| \sum_{j=k}^{(m+k)\wedge d} \binom{j-1}{k-1} \cdot \alpha_{m+k,j}(\lambda) \Big| \\ &\leq \sum_{j=k}^{(m+k)\wedge d} \binom{j-1}{k-1} \cdot |\alpha_{m+k,j}(\lambda)|. \end{aligned}$$

Plugging in the bound on $|\alpha_{m+k,j}(\lambda)|$ from Lemma 6 in the above, we get that

$$\begin{aligned} |\beta_{m,k}| &\leq \sum_{j=k}^{d} \binom{j-1}{k-1} \cdot \Big(\frac{4e \max\{m+k, d\}}{d}\Big)^d \\ &\overset{(i)}{\leq} \binom{d}{k} \cdot \Big(\frac{4e \max\{m+k, d\}}{d}\Big)^d \\ &\overset{(ii)}{\leq} \Big(\frac{8e \max\{m+k, d\}}{d}\Big)^d, \end{aligned}$$

where the inequality in (i) is given by the fact that any N and n, we have $\sum_{j=n}^{N} \binom{j}{n} = \binom{N+1}{n+1}$, and the inequality in (ii) holds because for any $k \leq d$, $\binom{d}{k} \leq 2^d$. □

B.3 Extension of the Cayley-Hamilton theorem

The following result is an extension of the Cayley-Hamilton theorem (Lemma 4) for rank d matrices, and relies on the coefficients $\beta_{m,k}$ defined above.

Lemma 9 (Cayley-Hamilton Theorem extension)**.** *Let $A \in \mathbb{C}^{N \times N}$ be a matrix with rank at most d, where $d \leq N$, and let $\lambda = (\lambda_1, \ldots, \lambda_d) \in \mathbb{C}^d$ denote the set of eigenvalues of A. Then, for any $m \geq 0$,*

$$A^{d+m+1} = \sum_{k=1}^{d} (-1)^{k+1} \beta_{m,k}(\lambda) A^{d+1-k} \tag{9}$$

where the coefficients vector $\beta_m(\lambda) := (\beta_{m,1}(\lambda), \ldots, \beta_{m,k}(\lambda))$ are given in Definition 2.

Proof. We give a proof by induction over m. For the base case, when $m = 0$, Lemma 4 implies that

$$A^{d+1} = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) A^{d+1-k} = \sum_{k=1}^{d} (-1)^{k+1} \beta_{0,k}(\lambda) A^{d+1-k}, \tag{10}$$

where the second equality follows form the definition of the vector $\beta_0(\lambda)$. We next prove the induction step.

Assume that the relation (9) holds for all $m' < m$. We note that

$$\begin{aligned} A^{d+1+m} &= AA^{d+1+(m-1)} \\ &\overset{(i)}{=} A\Big(\sum_{k=1}^{d}(-1)^{k+1}\beta_{m-1,k}(\lambda)\cdot A^{d+1-k}\Big) \\ &= \beta_{m-1,1}(\lambda)A^{d+1} + \sum_{k=2}^{n}(-1)^{k+1}\beta_{m-1,k}(\lambda)\cdot A^{d+2-k} \\ &= \beta_{m-1,1}(\lambda)A^{d+1} + \sum_{k=1}^{n-1}(-1)^{k}\beta_{m-1,k+1}(\lambda)\cdot A^{d+1-k}. \end{aligned}$$

where the equality in (i) following from using the relation (9) for time step $m-1$. Plugging in the expansion for A^{d+1} from (10) in the above, we get that

$$\begin{aligned} A^{d+1+m} &= \beta_{m-1,1}(\lambda)\Big(\sum_{k=1}^{d}(-1)^{k+1}\beta_{0,k}(\lambda)\cdot A^{d+1-k}\Big) + \sum_{k=1}^{d-1}(-1)^{k}\beta_{m-1,k+1}(\lambda)\cdot A^{d+1-k} \\ &= \sum_{k=1}^{d}(-1)^{k+1}(\beta_{m-1,1}(\lambda)\cdot\beta_{0,k}(\lambda) - \beta_{m-1,k+1}(\lambda))A^{d+1-k}. \end{aligned} \tag{11}$$

where in the second line, we defined $\beta_{m-1,d+1} = 0$. We next note that for any $k \in [d]$,

$$\begin{aligned} \beta_{m-1,1}(\lambda)\cdot\beta_{0,k}(\lambda) - \beta_{m-1,k+1}(\lambda) &= \beta_{m-1,1}(\lambda)\cdot\alpha_k(\lambda) - \beta_{m-1,k+1}(\lambda) \\ &= \beta_{m,k}(\lambda), \end{aligned}$$

where the second line above follows from the definition of $\beta_{m,k}$. Using this relation in (10), we get that

$$A^{d+1+m} = \sum_{k=1}^{d}(-1)^{k+1}\beta_{m,k}(\lambda)A^{d+1-k},$$

hence completing the induction step. Thus, the relation (9) holds for all $m \geq 0$. □

C Missing proofs from Section 4

C.1 Proof of Lemma 1

Lemma 1 (Autoregression on expected rewards)**.** *Let π be any policy for which the transition matrix T^π has rank at most d. Then, for any time step $h \geq d+1$, the expected reward for policy π at time step h, denoted by R_h^π, satisfies the auto-regression*

$$R_h^\pi = \sum_{k=1}^{d}(-1)^{k+1}\alpha_k(\lambda^\pi)R_{h-k}^\pi, \tag{6}$$

where $\lambda^\pi \in \mathbb{C}^d$ denotes the set of eigenvalues of the matrix T^π, and $\alpha_k(\lambda^\pi)$ is as defined in (3).

Proof. For any time step $h \geq 1$, let μ_h^π denote the distribution over the observation space $\mathcal{X}$ at time step h when starting from the initial distribution μ_0 and taking actions according to the policy π. Using the definition of the transition matrix T^π, we note that

$$\mu_h^\pi = T^\pi\mu_{h-1}^\pi, \tag{12}$$

where μ_0^π is defined as μ_0. Further, let $\nu^\pi \in \mathbb{R}^{\mathcal{X}}$ denotes the vector of expected rewards under policy π on the observation space, i.e., for any observation $x \in \mathcal{X}$,

$$\nu^\pi(x) := \mathbb{E}_{a\sim\pi(x)}[r(s,a)].$$

Thus, for any $h \leq H$, the expected reward R_h^π is given by the expression

$$R_h^\pi = \langle \nu^\pi, \mu_h^\pi \rangle = \langle \nu^\pi, (T^\pi)^{d+1} \mu_{h-d-1}^\pi \rangle, \tag{13}$$

where the second equality follows from recursively using the relation (12). Using the Cayley-Hamilton theorem (Lemma 4) for the matrix T^π, with rank at most d, we get that

$$(T^\pi)^{d+1} = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) (T^\pi)^{d+1-k},$$

where $\lambda = (\lambda_1^\pi, \ldots, \lambda_d^\pi)$ denotes the set of eigenvalues of T^π. Plugging the above in relation (13) for $h \geq d+1$, we get that

$$\begin{aligned} R_h^\pi &= \langle \nu^\pi, \sum_{k=1}^{d} \alpha_k(\lambda) (T^\pi)^{d+1-k} \mu_{h-d-1}^\pi \rangle \\ &= \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) \langle \nu^\pi, \mu_{h-k}^\pi \rangle = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) R_{h-k}^\pi, \end{aligned}$$

where the last equality follows from plugging back the expression for R_{h-k}^π from (13). □

C.2 Proof of Lemma 2

The following result provides an upper bound on the error in our estimates for the expected reward for any policy $\pi \in \Pi$.

Lemma 2 (Importance sampling). *For any $\delta \in (0,1)$, with probability at least $1-\delta$, for any policy $\pi \in \Pi$ and time step $h \in [3d]$, the estimates $\widehat{R}_h^\pi$ computed using importance sampling satisfy the error bound*

$$|\widehat{R}_h^\pi - R_h^\pi| \leq \sqrt{\frac{2K^{3d} \log(6d|\Pi|/\delta)}{n}} + \frac{2K^{3d} \log(6d|\Pi|/\delta)}{n}.$$

Proof. First fix any $h \in [3d]$ and $\pi \in \Pi$. The expected policy reward estimate is given by

$$\widehat{R}_h^\pi = \frac{1}{n} \sum_{i=1}^{n} r_h^t \prod_{h' \leq h} \left(K \mathbb{1}\{\pi(x_{h'}^t) = a_{h'}^t\} \right)$$

Clearly, $\widehat{R}_h^\pi$ is an unbiased estimate of R_h^π as

$$\begin{aligned} \mathbb{E}^{\bar{\pi}}[\widehat{R}_h^\pi] &= \frac{1}{n} \sum_{t=1}^{n} \mathbb{E}^{\bar{\pi}} \Big[r_h^t \prod_{h' \leq h} \left(K \mathbb{1}\{\pi(x_{h'}^t) = a_{h'}^t\} \right) \Big] \\ &= \frac{1}{n} \sum_{t=1}^{n} \mathbb{E}^{\bar{\pi}} \Big[r_h^t \prod_{h' \leq h} \frac{\pi(a_{h'}^t | x_{h'}^t)}{\bar{\pi}(a_{h'}^t | x_{h'}^t)} \Big] \\ &= \frac{1}{n} \sum_{t=1}^{n} \mathbb{E}^{\pi}[r_h^t] = \frac{1}{n} \sum_{t=1}^{n} R_h^\pi = R_h^\pi, \end{aligned}$$

where $\bar{\pi}$ denotes the stochastic policy that picks actions uniformly at random and is used to draw the trajectory $(x_h^t, a_h^t, r_h^t)_{h=1}^H$ for $t \in [n]$. The equality in the second line above follows from the definition of $\bar{\pi}$ and the last line follows by a change of measure to the case where the trajectories are sampled using the policy π. We next consider the second moment of each individual term in the estimator

$$\begin{aligned} \mathbb{E}^{\bar{\pi}} \Big[(r_h^t)^2 \prod_{h' \leq h} \left(K \mathbb{1}\{\pi(x_{h'}^t) = a_{h'}^t\} \right)^2 \Big] &\overset{(i)}{\leq} K^{2h} \prod_{h' \leq h} \mathbb{P}^{\bar{\pi}} \left(\pi(x_{h'}^t) = a_{h'}^t \right) \\ &\overset{(ii)}{=} K^{2h} \cdot \frac{1}{K^h} = K^h, \end{aligned}$$

where the inequality (i) uses that $0 \le r_h^i \le 1$, and the inequality (ii) holds because $\bar{\pi}$ draws actions uniformly at random which implies that $\mathbb{P}^{\bar{\pi}}\big(a_{h'}^i = \pi(x_{h'}^i) \mid x_{h'}^i\big) = 1/K$. Therefore the variance for the tth sample,

$$\mathbb{V}^{\bar{\pi}}\Big[r_h^t \prod_{h' \le h} \big(K \mathbb{1}\{\pi(x_{h'}^t) = a_{h'}^t\}\big)\Big] \le K^h.$$

Since all episodes are i.i.d., an application of Bernstein's inequality implies that with probability at least $1-\delta$

$$\begin{aligned}\left|\widehat{R}_h^\pi - R_h^\pi\right| &\le \sqrt{\frac{2\mathbb{V}^{\bar{\pi}}\left[K^h \mathbb{1}\left\{a_{1:h}^i = \pi(a^i)_{1:h}\right\} r_h^i\right] \log(2/\delta)}{n}} + \frac{4K^h}{3}\frac{\log(2/\delta)}{n} \\ &\le \sqrt{\frac{2K^h \log(2/\delta)}{n}} + \frac{2K^h \log(2/\delta)}{n}.\end{aligned}$$

Taking a union bound, we get that with probability at least $1-\delta$, for all $h \in [3d]$ and $\pi \in \Pi$,

$$\left|\widehat{R}_h^\pi - R_h^\pi\right| \le \sqrt{\frac{2K^h \log(6d|\Pi|/\delta)}{n}} + \frac{2K^h \log(6d|\Pi|/\delta)}{n}.$$

□

C.3 Proof of Lemma 3

Before providing the proof of Lemma 3, we first introduce the matrix $P(\lambda)$ that depends on the eigenvalues $\lambda \in \mathbb{C}^d$, and establish a technical result about the eigenspectrum of $P(\lambda)$.

Definition 3. *For any $d \ge 1$, $\lambda = (\lambda_1, \ldots, \lambda_d) \in \mathbb{C}^d$, define the matrix $P(\lambda) \in \mathbb{C}^{d \times d}$ such that*

$$[P(\lambda)]_{i,k} = \begin{cases} (-1)^{k+1}\alpha_k(\lambda) & \text{when } i = 1 \text{ and } 1 \le k \le d \\ 1 & \text{when } 2 \le i \le d-1 \text{ and } k = i-1 \\ 0 & \text{otherwise} \end{cases},$$

where the value of $\alpha_k(\lambda)$ is given in Definition 1.

The following technical result considers the eigenspectrum of the matrix $P(\lambda)$.

Lemma 10. *For any $\lambda \in \mathbb{C}^d$, the eigenvalues of the matrix $P(\lambda)$ are given by $(\lambda_1, \ldots, \lambda_d)$.*

Proof. For the ease of notation, define α_k to denote $\alpha_{k,k}(\lambda)$ for $k \in [d]$. We start by computing the characteristic polynomial of the matrix $P(\lambda)$, which is given by

$$\det(zI - P(\lambda)) = \det\begin{bmatrix} (z-\alpha_1) & \alpha_2 & -\alpha_3 & \cdots & (-1)^d\alpha_d \\ -1 & z & 0 & \cdots & 0 \\ 0 & -1 & z & \cdots & 0 \\ & \vdots & & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & z \end{bmatrix}.$$

Computing the determinant by expansing along the first row, we get that

$$\begin{aligned}\det(zI - P(\lambda)) &= (z-\alpha_1)z^{d-1} + \sum_{k=2}^{d}(-1)^{k+1} \cdot ((-1)^k\alpha_k) \cdot (-1)^{k-1} \cdot z^{d-k} \\ &= z^d - \alpha_1 z^{d-1} + \alpha_k z^{d-2} + \ldots + (-1)^d\alpha_d.\end{aligned}$$

Using the definition of α_k from Definition 1, we can factorize the above polynomial as

$$\det(zI - P(\lambda)) = \prod_{k=1}^{d}(z - \lambda_k).$$

Since, the eigenvalues of any matrix are given by the roots of its characteristic polynomial, the above computation shows that the eigenvalues of the matrix $P(\lambda)$ are given by $(\lambda_1, \ldots, \lambda_d)$. □

The following structural lemma shows that for any autoregression with coefficients $(\alpha_1(\lambda), \ldots, \alpha_k(\lambda))$, the $(m+d)$-th term can be expressed using the m-th power of the matrix $P(\lambda)$. Recall that the expected rewards for any policy satisfy such an autoregression whenever the underlying MDP has low rank (see Lemma 1).

Lemma 11. *Let $\lambda \in \mathbb{C}^d$ and $R_1, \ldots, R_d \in \mathbb{R}$. For any $h \geq d+1$, let R_h be given by*

$$R_h = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) R_{h-k}, \tag{14}$$

where the coefficient $\alpha_k(\lambda)$ are defined in Definition 1. Then, for any $m \geq 0$,

$$R_{m+d} = \langle U, P(\lambda)^m V \rangle \tag{15}$$

where the vector $U := (1, 0, \ldots, 0)^\top \in \mathbb{R}^d$, the vector $V := (R_d, R_{d-1}, \ldots, R_1)^\top \in \mathbb{R}^d$ and the matrix $P(\lambda) \in \mathbb{C}^d$ is defined in Definition 3.

Proof. For any $h \geq d$, define the vector $U_h \in \mathbb{R}^d$ such that

$$U_h := (R_h, R_{h-1}, \ldots, R_{h-d+1})^\top.$$

We first note that for any $j \in [d]$ such that $j \geq 2$,

$$U_h[j] = R_{h-(j)-1} = U_{h-1}[j-1].$$

Further, using the recurrence relation (14), we get that for any $h \geq d+1$,

$$U_h[1] = R_h = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) R_{h-k} = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) U_{h-1}[k].$$

The above two relation imply that for any $h \geq d+1$,

$$U_h = P(\lambda) U_{h-1}, \tag{16}$$

where the matrix $P(\lambda)$ is defined such that

$$[P(\lambda)]_{j,k} = \begin{cases} (-1)^{k+1}\alpha_k & \text{when } j = 1 \text{ and } 1 \leq k \leq d \\ 1 & \text{when } 2 \leq j \leq d \text{ and } k = j+1 \\ 0 & \text{otherwise} \end{cases}.$$

Setting $h = d + m$ in relation (16), we get that for any m

$$U_{m+d} = P(\lambda) U_{m-1+d} = \cdots = P(\lambda)^m U_d.$$

Finally, we note that for any $m \geq 0$,

$$R_{m+d} = \langle V, U_{m+d} \rangle = \langle V, P(\lambda)^m U_d \rangle,$$

where the vector $V = (1, 0, \ldots, 0) \in \mathbb{R}^d$ and the vector $U_d = (R_d, \ldots, R_1)^\top \in \mathbb{R}^d$. □

We are finally ready to prove Lemma 3. The following proof is based on the extension of the Cayley-Hamilton theorem for rank d matrices (see Lemma 9) and uses Lemma 11.

Lemma 3 (Error propagation bound)**.** *Let $\lambda, \widehat{\lambda} \in \mathbb{C}^d$ be such that $\max\{|\lambda_1|, |\widehat{\lambda}_1|\} \leq 1$. Further, with the initial values $R_1, \ldots, R_d$ and $\widetilde{R}_1, \ldots, \widetilde{R}_d$, let the sequence $\{R_h\}$ and $\{\widetilde{R}_h\}$ be given by*

$$R_h = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) R_{h-k} \qquad \text{and} \qquad \widetilde{R}_h = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}) \widetilde{R}_{h-k},$$

where the coefficients $\alpha_k(\lambda)$ and $\alpha_k(\widehat{\lambda})$ are define as in (3). Then, for all $h \geq 3d+1$,

$$|\widetilde{R}_h - R_h| \leq 2d \cdot \left(\frac{16eh}{d}\right)^{2d} \cdot \max_{h' \leq 3d} |R_{h'} - \widetilde{R}_{h'}|.$$

Proof. Using Lemma 11 for the sequences $\{R_h\}$ and $\{\widetilde{R}_h\}$ respectively, we get that for any $m \geq 0$,

$$R_{d+m} = \langle U, P(\lambda)^m V\rangle \qquad \text{and} \qquad \widetilde{R}_{d+m} = \langle \widetilde{U}, P(\widehat{\lambda})^m \widetilde{V}\rangle,$$

where the matrices $P(\lambda), P(\widehat{\lambda}) \in \mathbb{R}^{d\times d}$ are defined according to Definition 3 and the vectors $U, \widetilde{U}, V, \widetilde{V} \in \mathbb{R}^d$ are independent of λ and m. Thus, for any $m \geq 0$,

$$\begin{aligned} |R_{m+d} - \widetilde{R}_{m+d}| &= |\langle U, P(\lambda)^m V\rangle - \langle \widetilde{U}, P(\widehat{\lambda})^m V\rangle| \\ &= |\langle \bar{U}, \bar{P}^m \bar{V}\rangle|, \end{aligned} \tag{17}$$

where the vectors $\bar{U}, \bar{V} \in \mathbb{R}^{2d}$ and the block diagonal matrix $\bar{P} \in \mathbb{R}^{2d\times 2d}$ are defined as

$$\bar{V} := \begin{bmatrix} V \\ -\widetilde{V}\end{bmatrix}, \quad \bar{U} := \begin{bmatrix} U \\ -\widetilde{U}\end{bmatrix} \quad \text{and} \quad \bar{P} := \begin{bmatrix} P(\lambda) & 0 \\ 0 & P(\widehat{\lambda})\end{bmatrix}.$$

An application of Lemma 10 implies that the eigenvalues of the matrix $P(\lambda)$ and the matrix $P(\widehat{\lambda})$ are given by λ and $\widehat{\lambda}$ respectively. Since the matrix $\bar{P}$ is block-diagonal, we note that the set of eigenvalues of the matrix $\bar{P}$ is given by $\bar{\lambda} = (\lambda_1, \widehat{\lambda}_1, \ldots, \lambda_d, \widehat{\lambda}_d)$. Note that the vector $\bar{\lambda}$ is not sorted except for the first two coordinates, however $|\bar{\lambda}_k| \leq 1$ for all $k \in [2d]$. Using Lemma 9 for $2d \times 2d$ matrix $\bar{P}$, we get that for any $m \geq 2d+1$,

$$\bar{P}^{2d+m+1} = \sum_{k=1}^{2d} \beta_{m,k}(\bar{\lambda}) \cdot \bar{P}^{2d+1-k}.$$

Using the above relation with (17) and setting $m = h - 3d - 1$, we get that for any $h \geq 3d+1$,

$$\begin{aligned} |R_h - \widetilde{R}_h| = \left|\langle \bar{U}, \bar{P}^{h-d}\bar{V}\rangle\right| &= \left|\Big\langle \bar{U}, \sum_{k=1}^{2d} \beta_{h-3d-1,k}(\bar{\lambda}) \cdot \bar{P}^{2d+1-k}\bar{V}\Big\rangle\right| \\ &= \left|\sum_{k=1}^{2d} \langle \bar{U}, \beta_{h-3d-1,k}(\bar{\lambda}) \cdot \bar{P}^{2d+1-k}\bar{V}\rangle\right|. \end{aligned}$$

Using the triangle inequality on the right-hand side in the above, we obtain:

$$\begin{aligned} |R_h - \widetilde{R}_h| &\leq \sum_{k=1}^{2d} \left|\beta_{h-3d-1,k}(\bar{\lambda})\right| \cdot \left|\langle \bar{U}, \bar{P}^{2d+1-k}\bar{V}\rangle\right| \\ &\overset{(i)}{=} \sum_{k=1}^{2d} |\beta_{h-3d-1,k}(\bar{\lambda})| \cdot |R_{3d+1-k} - \widetilde{R}_{3d+1-k}| \\ &\overset{(ii)}{\leq} 2d \cdot \Big(\frac{4e\max\{h-3d-1+k, 2d\}}{d}\Big)^{2d} \cdot \max_{h'\leq 3d} |R_{h'} - \widetilde{R}_{h'}| \\ &\leq 2d \cdot \Big(\frac{16e\max\{h, d\}}{d}\Big)^{2d} \cdot \max_{h'\leq 3d} |R_{h'} - \widetilde{R}_{h'}|, \\ &= 2d \cdot \Big(\frac{16eh}{d}\Big)^{2d} \cdot \max_{h'\leq 3d} |R_{h'} - \widetilde{R}_{h'}|, \end{aligned}$$

where the equality in (i) holds due to relation (17) and the inequality (ii) is given by the bound on $|\beta_{h-3d-1,k}(\bar{\lambda})|$ from Lemma 8. The last line is due to the fact that $h > 3d$. □

C.4 Supporting technical results for the proof of Theorem 1

Lemma 12. *Let $\lambda \in \mathbb{C}^d$ be such that $|\lambda_k| \leq 1$ for all $k \in [d]$. Using the initial values $R_1, \ldots, R_d$, let R_h be defined as*

$$R_h := \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) R_{h-k}. \tag{18}$$

Further, let $\widehat{R}_1, \widehat{R}_2, \ldots, \widehat{R}_{3d}$ denote the estimates for $R_1, \ldots, R_{3d}$ respectively, such that

$$\max_{h\leq 3d} |\widehat{R}_h - R_h| \leq \eta. \tag{19}$$

Then,

(a) The optimization problem (4) in Algorithm 2 has a solution $(\widehat{\lambda}, \widehat{\Delta})$ such that

$$|\widehat{\Delta}| \leq 2d \cdot 4^d \cdot \eta.$$

(b) Further, let $\widetilde{R}_h$ be predictions according to line 5 in Algorithm 2 using the solution $\widehat{\lambda}$. Then,

$$\max_{h \leq 3d} |\widetilde{R}_h - R_h| \leq 2d \cdot (64e)^d \cdot \eta.$$

Proof. We prove the two parts separately below.

(a) We first show that the optimization problem in (4) is feasible. Specifically, we show that there exists a tuple (λ', Δ') that satisfies all the constraints in (4) such that $|\Delta'| \leq 2d4^d\eta$. Set $\lambda' = \lambda$. We note that $|\lambda'_1| = 1$ and $|\lambda'_k| \leq 1$ for all $k \leq d$ and thus all the constraints in (4) are satisfied. Furthermore, for any $h \leq 3d$,

$$\begin{aligned}
|\sum_{k=1}^{d}(-1)^{k+1}\alpha_k(\lambda)\widehat{R}_{h-k} - \widehat{R}_h| &\overset{(i)}{=} |\sum_{k=1}^{d}(-1)^{k+1}\alpha_k(\lambda)(\widehat{R}_{h-k} - R_{h-k}) - (\widehat{R}_h - R_h)| \\
&\overset{(ii)}{\leq} \sum_{k=1}^{d}|\alpha_k(\lambda)| \cdot |\widehat{R}_{h-k} - R_{h-k}| + |\widehat{R}_h - R_h| \\
&\overset{(iii)}{\leq} d \cdot 4^d \cdot \eta + \eta \\
&\leq 2d \cdot 4^d \cdot \eta
\end{aligned} \tag{20}$$

where the equality (i) follows from the relation (18) and the inequality (ii) follows from Triangle inequality. The inequality (iii) follows by plugging in the bound from Lemma 6 for $|\alpha_k(\lambda)|$ and using the bound in (19). The above implies that $|\Delta'| \leq 2d4^d\eta$.

Thus, any solution $(\widehat{\lambda}, \widehat{\Delta})$ of the optimization problem in (4) must satisfy

$$|\widehat{\Delta}| \leq 2d \cdot 4^d \cdot \eta. \tag{21}$$

(b) Let us first define some additional notation. For any $m \leq 2d$, define Δ_m as the error for mth expected reward when plugging in the minimizer solution $\widehat{\lambda}$, i.e.,

$$\widehat{\Delta}_m := \sum_{k=1}^{d}(-1)^{k+1}\alpha_k(\widehat{\lambda}) \cdot \widehat{R}_{d+m-k} - \widehat{R}_{d+m}. \tag{22}$$

Further, define Z_m as the error in our prediction for the expected reward at $(m+d)$th time step, i.e.

$$Z_m := \widetilde{R}_{m+d} - \widehat{R}_{m+d}. \tag{23}$$

In the following, we will show that for all $m \geq 1$,

$$Z_m = \widehat{\Delta}_m + \sum_{i=1}^{m-1}\beta_{i-1,1}(\widehat{\lambda}) \cdot \widehat{\Delta}_{m-i}, \tag{24}$$

where the coefficients $\beta_{i-1,1}$ are given in Definition 2.

Our desired bound follows as a direct consequence of (24). For any $1 \leq m \leq 2d$,

$$\begin{aligned}
|\widetilde{R}_{m+d} - R_{m+d}| &\leq |\widetilde{R}_{m+d} - \widehat{R}_{m+d}| + |\widehat{R}_{m+d} - R_{m+d}| \\
&\overset{(i)}{\leq} |\widehat{\Delta}_m + \sum_{i=1}^{m-1}\beta_{i-1,1}(\widehat{\lambda}) \cdot \widehat{\Delta}_{m-i}| + \eta \\
&\overset{(ii)}{\leq} |\widehat{\Delta}_m| + \sum_{i=1}^{m-1}|\beta_{i-1,1}(\widehat{\lambda})||\widehat{\Delta}_{m-i}| + \eta
\end{aligned}$$

$$\overset{(iii)}{\leq} |\widehat{\Delta}| + \sum_{i=1}^{m-1} |\beta_{i-1,1}(\widehat{\lambda})||\widehat{\Delta}| + \eta$$
$$\overset{(iv)}{\leq} 2d \cdot 4^d \cdot \Big(\frac{8e \max\{m,d\}}{d}\Big)^d \cdot \eta$$
$$\leq 2d \cdot (64e)^d \cdot \eta,$$

where the inequality (i) follows from the definition of Z_m in (23) and by using the bound in (19), and the inequality (ii) above is due to Triangle inequality. The inequality (iii) above follows by using the fact that $|\widehat{\Delta}_m| \leq |\widehat{\Delta}|$ for all $m \leq 2d$. Finally, the inequality (iv) follows by plugging in the bound in (21) and by using Lemma 8 to bound $|\beta_{i-1,1}(\widehat{\lambda})|$.

Proof of relation (24). We prove this by induction over m. For the base case, when $m = 1$,

$$Z_{d+1} = \widetilde{R}_{d+1} - \widehat{R}_{d+1} \overset{(i)}{=} \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}) \cdot \widehat{R}_k - \widehat{R}_{d+1} \overset{(ii)}{=} \widehat{\Delta}_1,$$

where the equality in (i) follows from the definition of $\widetilde{R}_{d+1}$ holds due to (18) and (ii) follows from the definition of $\widehat{\Delta}_1$.

We next show the induction step. For any $m \geq 2$, suppose that the relation (24) holds for all times $m' < m$. We note that

$$\begin{aligned}
Z_{d+m} &= \widetilde{R}_{d+m} - \widehat{R}_{d+m} \\
&\overset{(i)}{=} \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}) \cdot \widetilde{R}_{d+m-k} - \widehat{R}_{d+m} \\
&= \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}) \cdot \widetilde{R}_{d+m-k} - \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}) \cdot \widehat{R}_{d+m-k} \\
&\qquad + \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}) \cdot \widehat{R}_{d+m-k} - \widehat{R}_{d+m} \\
&\overset{(ii)}{=} \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}) \cdot \widetilde{R}_{d+m-k} - \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}) \cdot \widehat{R}_{d+m-k} + \widehat{\Delta}_m \\
&= \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}) \cdot \Big(\widetilde{R}_{d+m-k} - \widehat{R}_{d+m-k}\Big) + \widehat{\Delta}_m \\
&\overset{(iii)}{=} \sum_{k=1}^{d} (-1)^{k+1} \beta_{0,k}(\widehat{\lambda}) \cdot \Big(\widetilde{R}_{d+m-k} - \widehat{R}_{d+m-k}\Big) + \widehat{\Delta}_m \\
&\overset{(iv)}{=} \sum_{k=1}^{d} (-1)^{k+1} \beta_{0,k} \cdot Z_{m-k} + \widehat{\Delta}_m.
\end{aligned}$$

where (i) follows from the definition of $\widetilde{R}_{d+m}$ (see (5)) and (ii) follows by the definition of $\widehat{\Delta}_m$ in (22). The equality (iii) above is due to the fact that $\beta_{0,k}(\widehat{\lambda}) = \alpha_k(\widehat{\lambda})$ (by definition) and finally, the equality (iv) follows from the definition of Z_{m-k} in (23). Plugging in the induction hypothesis for Z_{m-k} in the above, we get that

$$\begin{aligned}
Z_{d+m} &= \sum_{k=1}^{d} (-1)^{k+1} \beta_{0,k}(\widehat{\lambda}) \cdot \Big(\widehat{\Delta}_{m-k} + \sum_{j=1}^{m-k-1} \beta_{j-1,1}(\widehat{\lambda}) \cdot \widehat{\Delta}_{m-k-j}\Big) + \widehat{\Delta}_m \\
&= \widehat{\Delta}_m + \sum_{i=1}^{m-1} \widehat{\Delta}_{m-i} \cdot \Big((-1)^{i+1} \beta_{0,i}(\widehat{\lambda}) + \sum_{j=1}^{i-1} (-1)^{i-j-1} \beta_{j-1,1}(\widehat{\lambda}) \cdot \beta_{0,i-j}\Big)
\end{aligned}$$

$$= \widehat{\Delta}_m + \sum_{i=1}^{m-1} \beta_{i-1,1}(\widehat{\lambda}) \cdot \widehat{\Delta}_{m-i},$$

where the second line above follows by rearranging the terms and using the fact that $\beta_{0,k}(\widehat{\lambda}) = 0$ whenever $k > d$, and the equality in the last line holds by using the fact that $\beta_{0,k}(\widehat{\lambda}) \cdot \beta_{h-1,1}(\widehat{\lambda}) = \beta_{h-1,k+1}(\widehat{\lambda}) + \beta_{h,k}(\widehat{\lambda})$ for all $h, k \geq 0$ (see Definition 2). This completes the induction step, hence proving (24) for all $m \geq 1$. □

C.5 Proof of Theorem 1

We finally provide the proof of Theorem 1 that characterizes the performance guarantee for the policy $\widetilde{\pi}$ returned by Algorithm 1.

Proof of Theorem 1 . Starting from Lemma 2, we get that with probability at least $1 - \delta$, for every policy $\pi \in \Pi$, our estimate $\widehat{R}^{\pi}_h$ computed in line 3 of Algorithm 2 satisfies the error bound

$$\max_{h' \leq 3d} |\widehat{R}^{\pi}_{h'} - R^{\pi}_{h'}| \leq \min\left\{ \sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}}, \frac{4K^{3d}\log(6d|\Pi|/\delta)}{n} \right\}. \tag{25}$$

Now, consider any policy $\pi \in \Pi$, and let $\widehat{\lambda}^{\pi}$, $\widehat{\Delta}^{\pi}$, $\widetilde{R}^{\pi}_h$ and $\widetilde{V}^{\pi}$ denote the corresponding local variables in the procedure ValEstimate when invoked in Algorithm 1 for the policy π. Further, let λ^{π} denote the eigenvalues of the transition matrix T^{π}. As a consequence of Lemma 1, the expected rewards R^{π}_h satisfy an autoregression where the coefficients are determined by λ^{π}. Specifically, for any $h \geq d + 1$,

$$R^{\pi}_h = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda^{\pi}) \cdot R^{\pi}_{h-k}.$$

Furthermore, by definition (see line 5 of Algorithm 2), the predicted rewards $\widetilde{R}^{\pi}_h$ also satisfy a similar autoregression where the coefficients are determined by $\widehat{\lambda}^{\pi}$, the solution of the optimization problem in (4) for the policy π. We have, for any $h \geq d + 1$,

$$\widetilde{R}^{\pi}_h = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}^{\pi}) \cdot \widetilde{R}^{\pi}_{h-k}$$

where $\widetilde{R}_{h'} := \widehat{R}_{h'}$ for $h' \leq d$. Additionally, also note that T^{π} is a stochastic matrix and thus $|\lambda^{\pi}_k| \leq 1$ for all $k \in [d]$. By definition, we also have that $|\widehat{\lambda}^{\pi}_k| \leq 1$. Thus, using the error propagation bound in Lemma 3 for the sequences $\{R^{\pi}_h\}$ and $\{\widetilde{R}^{\pi}_h\}$ we get that for any $h \geq 3d + 1$,

$$|\widetilde{R}^{\pi}_h - R^{\pi}_h| \leq 2d \cdot \left(\frac{16eh}{d}\right)^{2d} \max_{h' \leq 3d} |\widetilde{R}_{h'} - R_{h'}|.$$

The above bound implies that for any $h \geq 1$,

$$|\widetilde{R}^{\pi}_h - R^{\pi}_h| \leq 2d \cdot \left(\frac{16e(h \vee d)}{d}\right)^{2d} \max_{h' \leq 3d} |\widetilde{R}_{h'} - R_{h'}| \tag{26}$$

We note that an application of Lemma 12 implies that the predicted rewards $\widetilde{R}^{\pi}_{h'}$ satisfy the error bound

$$\begin{aligned} \max_{h' \leq 3d} |\widetilde{R}^{\pi}_{h'} - R^{\pi}_{h'}| &\leq 2d \cdot (64e)^d \cdot \max_{h' \leq 3d} |\widehat{R}^{\pi}_{h'} - R^{\pi}_{h'}| \\ &\leq 2d \cdot (64e)^d \cdot \eta, \end{aligned}$$

where η denotes the right hand side of (25). Plugging the above in (26), we get that

$$|\widetilde{R}^{\pi}_h - R^{\pi}_h| \leq 4d^2 \cdot \left(\frac{128e^2(h \vee d)}{d}\right)^{2d} \cdot \eta. \tag{27}$$

for any $h \geq 1$. Thus, the error in the estimated value $\widetilde{V}^{\pi}$ for the policy π is bounded by

$$\begin{aligned}
|\widetilde{V}^{\pi} - V^{\pi}| &= |\sum_{h=1}^{H} (\widetilde{R}_h^{\pi} - R_h^{\pi})| \\
&\leq \sum_{h=1}^{H} |\widetilde{R}_h^{\pi} - R_h^{\pi}| \\
&\leq \sum_{h=1}^{H} 4d^2 \cdot \Big(\frac{128e^2(h \vee d)}{d}\Big)^{2d} \cdot \eta \\
&\leq 4d^3 \cdot \Big(\frac{128e^2 H}{d}\Big)^{2d} \cdot \eta, \qquad (28)
\end{aligned}$$

where the inequality in the second last line follows by using the bound in (27), and the inequality in the last line holds because $H \geq d$.

Since π is arbitrary in the above chain of arguments, the error bound in (28) holds for all policies $\pi \in \Pi$. Thus, for any $\pi \in \Pi$, the policy $\widetilde{\pi}$ returned in line 4 of Algorithm 1 satisfies

$$\begin{aligned}
V^{\widetilde{\pi}} - V^{\pi} &= (\widetilde{V}^{\pi} - V^{\pi}) + (\widetilde{V}^{\widetilde{\pi}} - \widetilde{V}^{\pi}) + (V^{\widetilde{\pi}} - \widetilde{V}^{\widetilde{\pi}}) \\
&\geq (\widetilde{V}^{\pi} - V^{\pi}) + (V^{\widetilde{\pi}} - \widetilde{V}^{\widetilde{\pi}}) \\
&\geq -|\widetilde{V}^{\pi} - V^{\pi}| - |V^{\widetilde{\pi}} - \widetilde{V}^{\widetilde{\pi}}|,
\end{aligned}$$

where the inequality in the second line follows from the fact that $\widetilde{V}^{\widetilde{\pi}} \geq \widetilde{V}^{\pi}$ for every $\pi \in \Pi$ by the definition of the policy $\widetilde{\pi}$. Using the bound from (28) for policies π and $\widetilde{\pi} \in \Pi$ in the above, we get that

$$\begin{aligned}
V^{\widetilde{\pi}} &\geq V^{\pi} - 4d^3 \cdot \Big(\frac{128e^2 H}{d}\Big)^{2d} \cdot \eta \\
&\geq V^{\pi} - 4d^3 \cdot \Big(\frac{128e^2 H}{d}\Big)^{2d} \cdot \min\Bigg\{\sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}}, \frac{4K^{3d}\log(6d|\Pi|/\delta)}{n}\Bigg\} \\
&\geq V^{\pi} - 4d^3 \cdot \Big(\frac{128e^2 H}{d}\Big)^{2d} \sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}}
\end{aligned}$$

where the inequality in the second line above follows by plugging in the value of η as the right hand side of (25), and the inequality in the last line holds due to the fact that $-\min\{a, b\} \geq -a$ for any $a, b \geq 0$.

Since the above holds for any $\pi \in \Pi$, we have that

$$V^{\widetilde{\pi}} \geq \max_{\pi \in \Pi} V^{\pi} - 4d^3 \cdot \Big(\frac{128e^2 H}{d}\Big)^{2d} \sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}},$$

hence proving the desired statement. □

D Adaptive upper bounds

In this section, we present Algorithm 3 whose performance guarantee adapts to the unknown eigenspectrum of the underlying transition matrix. We then proceed to the proof of our adaptive upper bound Theorem 3.

D.1 Adaptive policy search algorithm

Algorithm 3 Adaptive policy search algorithm (Adaptivity to unknown eigenspectrum)

Input: horizon H, rank d, number of episodes n, finite policy class Π

1: Collect the dataset $\mathcal{D} = \{(x_h^t, a_h^t, r_h^t)_{h=1}^H\}_{t=1}^n$ by sampling n trajectories where actions are sampled from Uniform($\mathcal{A}$).
2: **for** policy $\pi \in \Pi$ **do**
3: Estimate $\widetilde{V}^\pi$ by calling AdaValEstimate($H, d, \mathcal{D}, \pi$).
4: **Return:** policy $\widetilde{\pi}$ with best estimated value $\widetilde{\pi} \in \operatorname{argmax}_{\pi\in\Pi} \widetilde{V}^\pi$.

Algorithm 4 Adaptive value estimation by autoregressive extrapolation

1: **function** AdaValEstimate($H, d, \mathcal{D}, \pi$):
2: Set $\Delta = 2d4^d \min\Big\{\sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}}, \frac{4K^{3d}\log(6d|\Pi|/\delta)}{n}\Big\}$.
3: **for** time step $h = 1, \dots, 3d$ **do**
4: Estimate expected rewards by importance sampling

$$\widehat{R}_h = \frac{1}{n}\sum_{i=1}^n r_h^i \prod_{h'\le h} \big(K\mathbb{1}\{\pi(x_{h'}^i) = a_{h'}^i\}\big)$$

5: Estimate eigenvalues of the autoregression by solving the optimization problem:

$$\begin{aligned} \widehat{\lambda} \leftarrow \operatorname*{argmin}_{\lambda\in\mathbb{C}^d} \prod_{k=2}^d \Big(\sum_{h=0}^{H-1} |\lambda_k|^h\Big) & \qquad (29)\\ \text{s.t. } |\lambda_1| = 1, |\lambda_k| \le 1 & \qquad \text{for } 2 \le k \le d,\\ \Big|\sum_{k=1}^d (-1)^{k+1}\alpha_k(\lambda)\widehat{R}_{h-k} - \widehat{R}_h\Big| \le \Delta & \qquad \text{for } d+1 \le h \le 3d. \end{aligned}$$

6: Predict $\widetilde{R}_h$ as:

$$\widetilde{R}_h = \begin{cases} \widehat{R}_h & \text{for } 1 \le h \le d \\ \sum_{k=1}^d (-1)^{k+1}\alpha_k(\widehat{\lambda})\widetilde{R}_{h-k} & \text{for } d+1 \le h \le H \end{cases}.$$

7: **return:** Estimate of the value function $\widetilde{V} = \sum_{h=1}^H \widetilde{R}_h$.

D.2 Adaptive error propagation bound

The main technical innovation that leads to the adaptive upper bound in Theorem 3 is the following bound on the propagated error in the hth step prediction. The bound in (30) adapts to the eigenvalues λ and $\widehat{\lambda}$, which define the auto-regressions for $\{R_h\}$ and $\{\widetilde{R}_h\}$ respectively.

Lemma 13 (Adaptive error propagation bound)**.** *Let $\lambda, \widehat{\lambda} \in \mathbb{C}^d$ be such that $\max\{|\lambda_1|, |\widehat{\lambda}_1|\} \le 1$. Further, with the initial values $\{R_1, \dots, R_d\}$ and $\{\widetilde{R}_1, \dots, \widetilde{R}_d\}$, let the sequence $\{\widetilde{R}_h\}$ and $\{R_h\}$ be given by*

$$R_h = \sum_{k=1}^d (-1)^{k+1}\alpha_k(\lambda)\cdot R_{h-k} \qquad \textit{and} \qquad \widetilde{R}_h = \sum_{k=1}^d (-1)^{k+1}\alpha_k(\widehat{\lambda})\cdot \widetilde{R}_{h-k},$$

where the coefficients $\alpha_k(\widehat{\lambda})$ and $\alpha_k(\lambda)$ are given in Definition 1. Then, for all $h \geq 1$,

$$|\widetilde{R}_h - R_h| \leq 2^{2d} h \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\lambda_k|^j \Big) \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\widehat{\lambda}_k|^j \Big) \cdot \max_{h' \leq 3d} |R_{h'} - \widetilde{R}_{h'}|. \tag{30}$$

We defer the proof to Appendix D.2.2.

D.2.1 Supporting technical results for the proof of Lemma 13

Lemma 14. *Given any vectors $u, v \in \mathbb{R}^d$ and a diagonalizable matrix $A \in \mathbb{R}^{d \times d}$ with eigenvalues $\lambda_1, \ldots, \lambda_d$ such that $|\lambda_1| = 1$, let $z_{m,k}$ be defined such that*

$$z_{m,k} = \begin{cases} u^\top A^m v & \text{when} \quad k = 0 \\ z_{m,k-1} - \lambda_{d+1-k} \cdot z_{m-1,k-1} & \text{when} \quad 1 \leq k \leq d, \end{cases}.$$

where $k \leq \min\{d, m\}$. Then, $z_{m,d} = 0$ for all $m \geq d$. Furthermore, for any $k \leq d$, the following inequality holds:

$$|z_{d,k}| \leq 2^k \max_{1 \leq i \leq d} |u^\top A^i v|.$$

Proof. We prove the two statements separately below.

(a) We first show that $z_{m,d} = 0$ for all $m \geq d$. Since, the matrix A is diagonalizable, we have

$$A = Q \Lambda Q^{-1},$$

where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_d)$ and where $Q \in \mathbb{R}^{d \times d}$ is the matrix whose kth column is an eigenvector q_k corresponding to the eigenvalue λ_k. In order to prove this, we will show that for any $m \geq 0$ and $k \leq \min(d, m)$,

$$z_{m,k} = u^T Q D_{m,k} Q^{-1} v, \tag{31}$$

where the matrix $D_{m,k} \in \mathbb{R}^{d \times d}$ is diagonal with entries given by

$$[D_{m,k}]_{i,i} = \begin{cases} 0 & \text{if} \quad d+1-k \leq i \leq d \\ \lambda_i^{m-k} \prod_{k'=1}^{k} (\lambda_i - \lambda_{d+1-k'}) & \text{otherwise.} \end{cases} \tag{32}$$

Specifically, the diagonal entry $[D_{m,k}]_{i,i} = 0$ for $i \geq d+1-k$. Observe that for $m, k \geq 1$ and $i < d+1-k$, the following relation holds:

$$\begin{aligned} [D_{m,k}]_{i,i} &= \lambda_i^{m-k} \prod_{k'=1}^{k} (\lambda_i - \lambda_{d+1-k'}) \\ &= (\lambda_i - \lambda_{d+1-k}) \lambda_i^{m-k} \prod_{k'=1}^{k-1} (\lambda_i - \lambda_{d+1-k'}) \\ &= \lambda_i^{m-k+1} \prod_{k'=1}^{k-1} (\lambda_i - \lambda_{d+1-k'}) - \lambda_{d+1-k} \lambda_i^{m-k} \prod_{k'=1}^{k-1} (\lambda_i - \lambda_{d+1-k'}) \\ &= [D_{m,k-1}]_{i,i} - \lambda_{d+1-k} [D_{m-1,k-1}]_{i,i}. \end{aligned} \tag{33}$$

Also, for $m, k \geq 1$ and $i = d+1-k$, the following relation holds:

$$\begin{aligned} [D_{m,k}]_{i,i} &= 0 \\ &= \lambda_{d+1-k}^{m-k+1} \prod_{k'=1}^{k-1} (\lambda_i - \lambda_{d+1-k'}) - \lambda_{d+1-k} \lambda_{d+1-k}^{m-k} \prod_{k'=1}^{k-1} (\lambda_i - \lambda_{d+1-k'}) \\ &= [D_{m,k-1}]_{d+1-k,d+1-k} - \lambda_{d+1-k} [D_{m-1,k-1}]_{d+1-k,d+1-k}. \end{aligned} \tag{34}$$

For $i > d + 1 - k$, by definition, we have

$$[D_{m,k}]_{i,i} = [D_{m-1,k}]_{i,i} = [D_{m-1,k-1}]_{i,i} = 0. \tag{35}$$

We prove (31) by an induction over the set of tuples (m, k). The induction proceeds in a row-first manner by first keeping m fixed and increasing k from 1 to d; we then increase m to $m + 1$ and proceed with the next row in the set of tuples (m, k). For the base case, for $k = 0$ and any $m \geq 0$,

$$z_{m,0} = u^T A^m v \overset{(i)}{=} u^T Q \Lambda^m Q^{-1} v \overset{(ii)}{=} u^T Q D_{m,0} Q^{-1} v,$$

where the equality (i) follows by using the fact that $A = Q\Lambda Q^{-1}$ and the inequality in (ii) is given by the definition of the matrix $D_{m,0}$.

We next prove the induction step. For any $m \geq 0$ and $k \leq d$, suppose that (31) holds for every tuple (m', k') where $m' < m$ and $k' \leq \min(m', d)$, and for every tuple (m, k') where $0 \leq k' < k$. In the following, we will show that the relation (31) will hold for the tuple (m, k) as well. Using the definition of $z_{m,k}$, we get that

$$\begin{aligned}
z_{m,k} &= z_{m,k-1} - \lambda_{d+1-k}\, z_{m-1,k-1} \\
&= u^T Q D_{m,k-1} Q^{-1} v - \lambda_{d+1-k} u^T Q D_{m-1,k-1} Q^{-1} v && \text{(Equation 31)} \\
&= u^T Q (D_{m,k-1} - \lambda_{d+1-k} D_{m-1,k-1}) Q^{-1} v \\
&= u^T Q (D_{m,k}) Q^{-1} v. && \text{(Equations 33, 34, and 35)}
\end{aligned}$$

This completes the induction step, thereby proving that (31) holds for all $m \geq 0$ and $k \leq \min(d, m)$. Setting $k = d$ in relation (31) gives $D_{m,d} = \text{diag}(0, \ldots, 0)$ for any $m \geq d$ and thus the following:

$$z_{m,d} = u^T Q D_{m,d} Q^{-1} v = 0.$$

(b) In the following, we will show that for any $m \geq d$ and $k \leq \min\{d, m\}$,

$$|z_{m,k}| \leq 2^k \Delta, \tag{36}$$

where $\Delta := \max\{|u^T Av|, \ldots, |u^T A^d v|\}$.

We prove (36) by an induction over the set of tuples (m, k). The induction proceeds in a row-first manner by first keeping m fixed and increasing k from 1 to d; we then increase m to $m + 1$ and proceed with the next row in the set of tuples (m, k). For the base case, we note that for $k = 0$ and any $m \leq d$,

$$|z_{m,0}| \leq u^T A^m v \leq \max\{|u^T Av|, \ldots, |u^T A^d v|\} = \Delta.$$

We next show the induction step. Given any m and k such that $k \leq \min\{d, m\}$, assume that (36) holds for every tuple (m', k') where $m' < m$ and $k' \leq \min(m', d)$, and for every tuple (m, k') where $0 \leq k' < k$. In the following, we will show that the relation (36) holds for the tuple (m, k) as well. Using the definition of $z_{m,k}$, we get that

$$\begin{aligned}
|z_{m,k}| &= |z_{m,k-1} - \lambda_{d+1-k} \cdot z_{m-1,k-1}| \\
&\leq |z_{m,k-1}| + |\lambda_{d+1-k}||z_{m-1,k-1}| \\
&\leq |z_{m,k-1}| + |z_{m-1,k-1}|,
\end{aligned}$$

where the last line holds because $|\lambda_{d+1-k}| \leq 1$. Using the bound of (36) for the tuples $(m, k-1)$ and $(m-1, k-1)$, we obtain:

$$|z_{m,k}| \leq 2^{k-1}\Delta + 2^{k-1}\Delta \leq 2^k \Delta.$$

This completes the induction step, hence proving (36) for all $m \geq d$ and $k \leq \min\{d, m\}$.

Finally, setting $m = d$ in (36) gives us the desired result.

□

Lemma 15. *Let $A \in \mathbb{R}^{d\times d}$ be a matrix with eigenvalues $(\lambda_1, \ldots, \lambda_d)$ such that $|\lambda_1| = 1$ and $|\lambda_k| \leq 1$, for all $k \in [d]$. Then, for any two vectors $u \in \mathbb{R}^d$ and $v \in \mathbb{R}^d$ and any $m \geq d+1$, the following inequality holds:*

$$|u^T A^m v| \leq 2^d \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{m-d} |\lambda_k|^j \Big) \cdot \max\big\{ |u^T A v|, \ldots, |u^T A^d v| \big\}.$$

Proof. We will first prove the result when the matrix A has distinct eigenvalues. We will later extend the proof for general matrices A.

Simpler setting: When A has distinct eigenvalues. We first introduce some notation to be used in the proof. Fix the vectors $u, v \in \mathbb{R}^d$. For any $m \geq 1$, and $k \leq \min\{d, m\}$, define $z_{m,k} \in \mathbb{R}$ as follows:

$$z_{m,k} = \begin{cases} u^T A^m v & \text{when} \quad k = 0 \\ z_{m,k-1} - \lambda_{d+1-k} \cdot z_{m-1,k-1} & \text{when} \quad 1 \leq k \leq d. \end{cases} . \tag{37}$$

Further, define $\Delta := \max\big\{ |u^T A v|, \ldots, |u^T A^d v| \big\}$. Since the matrix A has distinct eigenvalues, A is diagonalizable and thus by Lemma 14, the following inequality holds for any $k \leq d$:

$$|z_{d,k}| \leq 2^k \Delta \leq 2^d \Delta. \tag{38}$$

In order to prove the desired result, we first show that for any $k \leq d-1$, and any $m \geq d$,

$$|z_{m,k}| \leq 2^d \Delta \cdot \prod_{k'=2}^{d-k} \Big(\sum_{j=0}^{m-d} |\lambda_{k'}|^j \Big). \tag{39}$$

(39) can be shown by induction over k. For the base case, when $k = d-1$, we note that for any $m > d$,

$$\begin{aligned} |z_{m,d-1}| &= |z_{m,d} + \lambda_1 \cdot z_{m-1,d-1}| \\ &= |\lambda_1||z_{m-1,d-1}| \\ &\leq |z_{m-1,d-1}|, \end{aligned}$$

where the first line follows from the definition of $z_{m,d}$, and the equality in the second line holds because $z_{m,d} = 0$ for all $m \geq d$ (see Lemma 14). The inequality in the last line above is given by the fact that $|\lambda_d| \leq 1$. Repeating the above $m-d$ times, we get that

$$|z_{m,d-1}| \leq |z_{d,d-1}| \leq 2^{d-1}\Delta,$$

where the second inequality above follows from the bound (38).

We next show the induction step. For any $k \leq d-2$, suppose (39) holds for all $k' > k$ and all $m \geq d$. In the following, we will show that (39) also holds for k. Using the definition of $z_{m,k+1}$ from (37), we obtain:

$$|z_{m,k}| = |z_{m,k+1} + \lambda_{d-k} \cdot z_{m-1,k}| \leq |z_{m,k+1}| + |\lambda_{d-k}||z_{m-1,k}|.$$

Reiterating the above $m-d$ times by upper-bounding $|z_{m-1,k}|$ yields:

$$|z_{m,k}| \leq \sum_{j=0}^{m-d-1} |\lambda_{d-k}|^j |z_{m-j,k+1}| + |\lambda_{d-k}|^{m-d}|z_{d,k}|.$$

Plugging in the bound (39) for $|z_{m-j,k+1}|$ and the bound (38) for $|z_{d,k}|$ in the above, we get that

$$\begin{aligned} |z_{m,k}| &\leq \sum_{j=0}^{m-d-1} |\lambda_{d-k}|^j 2^d \Delta \cdot \prod_{k'=2}^{d-k-1} \Big(\sum_{j'=0}^{m-j-d} |\lambda_{k'}|^{j'} \Big) + |\lambda_{d-k}|^{m-d} 2^d \Delta \\ &\leq 2^d \Delta \Big(\sum_{j=0}^{m-d-1} |\lambda_{d-k}|^j + |\lambda_{d-k}|^{m-d} \Big) \cdot \prod_{k'=2}^{d-k-1} \Big(\sum_{j'=0}^{m-d} |\lambda_{k'}|^{j'} \Big) \end{aligned}$$

$$= 2^d \Delta \cdot \prod_{k'=2}^{d-k} \Big(\sum_{j=0}^{m-d} |\lambda_{k'}|^j \Big),$$

where the inequality in the second line follows from the fact that $\sum_{j'=0}^{m-d} |\lambda_{k'}|^{j'} \geq 1$. This completes the induction step, thereby proving that (39) holds for all $k \leq d-1$. The final statement follows by setting $k = 0$ in (39).

Extension to general matrices A. We now prove the result for a general matrix A by using the fact that matrices with distinct eigenvalues are dense in the space of $d \times d$ matrices. From Theorem 5, we note that for every $\varepsilon > 0$, there exists a matrix B^ε with distinct eigenvalues, denoted by $\lambda^\varepsilon \in \mathbb{C}^d$, such that:

(a) $\|A^m - (B^\varepsilon)^m\| \leq \varepsilon$ for all $m \geq 1$.

(b) $|\lambda_1^\varepsilon| = 1$ and $|\lambda_k^\varepsilon| \leq 1$ for all $k \in [d]$.

(c) $\|B^\varepsilon\|_\infty \leq \|A\|_\infty$.

Using the above proof for the matrix B^ε which has distinct eigenvalues, we get that for all $m \geq d+1$,

$$|u^T (B^\varepsilon)^m v| \leq 2^d \prod_{k=2}^{d} \Big(\sum_{j=0}^{m-d} |\lambda_k^\varepsilon|^j \Big) \cdot \max\big\{ |u^T (B^\varepsilon) v|, \ldots, |u^T (B^\varepsilon)^d v| \big\}. \tag{40}$$

Furthermore, an application of Theorem 6 implies that the eigenvalues of the matrix A and B^ε are related as:

$$\begin{aligned} \max_j \min_i \ |\lambda_i - \lambda_j^\varepsilon| &\leq (\|A\| + \|B^\varepsilon\|)^{1-1/d} \|A - B\|^{1/d} \\ &\leq (d^2 \|A\|_\infty + d^2 \|B^\varepsilon\|_\infty)^{1-1/d} \|A - B\|^{1/d} \\ &\leq (2d^2 \|A\|_\infty)^{(1-1/d)} \cdot \varepsilon^{1/d}, \end{aligned}$$

where the inequality in the second line above follows from the fact that for any matrix B, $\|B\| \leq \|B\|_F \leq d^2 \|B\|_\infty$. The inequality in the third line above is given by the fact that $\|B^\varepsilon\|_\infty \leq \|A\|_\infty$. Thus, if $\varepsilon \leq \frac{1}{2d^2 \|A\|_\infty} \cdot \min_{\lambda_i \neq \lambda_j} |\lambda_i - \lambda_j|$, the above bound implies that the eigenvalues of B^ε are such that

$$|\lambda_k - \lambda_k^\varepsilon| \leq (2d^2 \|A\|_\infty)^{(1-1/d)} \cdot \varepsilon^{1/d}, \tag{41}$$

for all $k \in [d]$.

Finally, using the fact that $\|A^m - (B^\varepsilon)^m\| \leq \varepsilon$ for all $m \geq 1$ and the bound on the deviation in eigenvalues from (41) in the relation (40), and taking the limit as ε approaches 0, we get that,

$$|u^T A^m v| \leq 2^d \prod_{k=2}^{d} \Big(\sum_{j=0}^{m-d} |\lambda_k|^j \Big) \cdot \max\big\{ |u^T A v|, \ldots, |u^T A^d v| \big\}.$$

This completes the proof of the lemma for general $d \times d$ matrices A. □

Theorem 5 (Modification of Corollary 1 in Hartfiel [1992]; Theorem 1 in [Hartfiel, 1995])**.** *Let A be a $d \times d$ matrix with eigenvalues $\lambda \in \mathbb{C}^d$ such that $|\lambda_1| = 1$ and $|\lambda_k| \leq 1$ for all $k \in [d]$. Then, for every $\varepsilon > 0$, there exists a matrix B^ε such that:*

(a) B^ε has distinct eigenvalues.

(b) $\|A^m - (B^\varepsilon)^m\| \leq \varepsilon$ for all $m \geq 1$.

(c) $|\lambda_1(B^\varepsilon)| = 1$ and $|\lambda_k(B^\varepsilon)| \leq 1$ for all $k \in [d]$.

(d) $\|B^\varepsilon\|_\infty \leq \|A\|_\infty$.

Theorem 6 (Theorem 8.1.1. in Bhatia [2013])**.** *Let A, B be $d \times d$ with eigenvalues $\lambda_1, \ldots, \lambda_d$ and $\lambda'_1, \ldots, \lambda'_d$ respectively. Then,*

$$\max_j \min_i \ |\lambda_i - \lambda'_j| \leq (\|A\| + \|B\|)^{1-1/d} \|A - B\|^{1/d}.$$

D.2.2 Proof of Lemma 13

We are finally ready to prove the adaptive error propagation bound given in Lemma 13.

Proof of Lemma 13 . Using Lemma 11 for the sequences $\{R_h\}$ and $\{\widetilde{R}_h\}$ respectively, we get that for any $m \geq 0$,

$$R_{d+m} = \langle U, P(\lambda)^m V \rangle$$

and

$$\widetilde{R}_{d+m} = \langle \widetilde{U}, P(\widehat{\lambda})^m \widetilde{V} \rangle,$$

where the matrices $P(\lambda), P(\widehat{\lambda}) \in \mathbb{R}^{d\times d}$ are defined according to Definition 2 and the vectors $U, \widetilde{U}, V, \widetilde{V} \in \mathbb{R}^d$ are independent of λ and m. Thus, for any $m \geq 0$,

$$\begin{aligned} |R_{m+d} - \widetilde{R}_{m+d}| &= |\langle U, P(\lambda)^m V\rangle - \langle \widetilde{U}, P(\widehat{\lambda})^m V\rangle| \\ &= |\langle \bar{U}, \bar{P}^m \bar{V}\rangle|, \end{aligned} \tag{42}$$

where the vectors $\bar{U}, \bar{\beta} \in \mathbb{R}^{2d}$ and the block diagonal matrix $\bar{P} \in \mathbb{R}^{2d\times 2d}$ are defined as

$$\bar{V} := \begin{bmatrix} V \\ -\widetilde{V} \end{bmatrix}, \quad \bar{U} := \begin{bmatrix} U \\ -\widetilde{U} \end{bmatrix} \quad \text{and} \quad \bar{P} := \begin{bmatrix} P(\lambda) & 0 \\ 0 & P(\widehat{\lambda}) \end{bmatrix}.$$

An application of Lemma 10 implies that the eigenvalues of the matrix $P(\lambda)$ and the matrix $P(\widehat{\lambda})$ are given by λ and $\widehat{\lambda}$ respectively. Since the matrix $\bar{P}$ is block-diagonal, we note that the set of eigenvalues of the matrix $\bar{P}$ is given by $\bar{\lambda} = (\lambda_1, \widehat{\lambda}_1, \ldots, \lambda_d, \widehat{\lambda}_d)$. Note that the vector $\bar{\lambda}$ is not sorted except for the first two coordinates, however $|\bar{\lambda}_k| \leq 1$ for all $k \in [2d]$. Using Lemma 15 for the $2d \times 2d$ matrix $\bar{P}$ and the vectors $\bar{U}$ and $\bar{\beta}$, we get that for any $m \geq 2d+1$,

$$\begin{aligned} |\langle \bar{U}, \bar{P}^m \bar{V}\rangle| &\leq 2^{2d} \cdot \prod_{k=2}^{2d} \Big(\sum_{j=0}^{m-2d} |\bar{\lambda}_k|^j \Big) \cdot \max\{|\langle \bar{U}, \bar{P}\bar{V}\rangle|, \ldots, |\langle \bar{U}, \bar{P}^{2d}\bar{V}\rangle|\} \\ &\leq 2^{2d} \cdot m \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{m-1} |\lambda_k|^j \Big) \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{m-1} |\widehat{\lambda}_k|^j \Big) \cdot \max_{m' \leq 2d} |\langle \bar{U}, \bar{P}^{m'} \bar{V}\rangle|, \end{aligned} \tag{43}$$

where the inequality in the last line uses the fact that $|\lambda_k| \leq 1$ and $|\widehat{\lambda}_k| \leq 1$ for all $k \in [d]$, and from thus $\sum_{j=0}^{m-1} |\widehat{\lambda}_k|^j \leq m$. Using the bound (43) in the relation (42), we get that for any $h \geq 3d+1$,

$$\begin{aligned} |R_h - \widetilde{R}_h| &\leq |\langle \bar{U}, \bar{P}^{h-d} V\rangle| \\ &\leq 2^{2d} h \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\lambda_k|^j \Big) \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\widehat{\lambda}_k|^j \Big) \cdot \max_{m' \leq 2d} |\langle \bar{U}, \bar{P}^{m'} \bar{V}\rangle| \\ &\overset{(i)}{=} 2^{2d} h \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\lambda_k|^j \Big) \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\widehat{\lambda}_k|^j \Big) \cdot \max_{m' \leq 2d} |R_{d+m'} - \widetilde{R}_{d+m'}| \\ &\leq 2^{2d} h \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\lambda_k|^j \Big) \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\widehat{\lambda}_k|^j \Big) \cdot \max_{h' \leq 3d} |R_{h'} - \widetilde{R}_{h'}|, \end{aligned}$$

where the equality (i) follows due to relation (42). □

D.3 Proof of Theorem 3

Before delving into the proof of Theorem 3, we first note the following technical lemma which concerns with the feasability and properties of the solutions of optimization problem (29) in Algorithm 4.

Lemma 16. *Let $\lambda \in \mathbb{C}^d$ such that $|\lambda_k| \leq 1$ for all $k \in [d]$. Using the initial values $R_1, \ldots, R_d$, let R_h be defined as*

$$R_h = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda^\pi) R_{h-k}. \tag{44}$$

Further, let $\widehat{R}_1, \widehat{R}_2, \ldots, \widehat{R}_{3d}$ denote the estimates for $R_1, \ldots, R_{3d}$ respectively, such that

$$\max_{h \leq 3d} |\widehat{R}_h - R_h| \leq \eta, \tag{45}$$

where $\eta := \min\Big\{\sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}}, \frac{4K^{3d}\log(6d|\Pi|/\delta)}{n}\Big\}$. Then,

(a) The optimization problem (29) in Algorithm 4 has a solution $\widehat{\lambda} \in \mathbb{C}^d$ such that $|\widehat{\lambda}_1| = 1$ and

$$\prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\widehat{\lambda}_k|^j\Big) \leq \prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\lambda_k|^j\Big).$$

(b) Further, let $\widetilde{R}_h$ be predictions according to line 7 in Algorithm 4 using the solution $\widehat{\lambda}$. Then,

$$\max_{h' \leq 3d} |\widetilde{R}_h - R_h| \leq 2d \cdot (64e)^d \cdot \eta.$$

Proof. In the following, we provide the proof for part-(a) of the lemma. The proof of part-(b) follows exactly along the lines of a similar statement proven in Lemma 12.

Proof of part-(a). We prove this by showing that the vector $\lambda \in \mathbb{R}^d$ satisfies all the constraints of the optimization problem in (29). First note that $|\lambda_1| = 1$ and $|\lambda_k| \leq 1$ for all $k \leq d$, by definition. Furthermore, for any $h \leq 3d$,

$$\begin{aligned}
|\sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) \widehat{R}_{h-k} - \widehat{R}_h| &\overset{(i)}{=} |\sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda)(\widehat{R}_{h-k} - R_{h-k}) - (\widehat{R}_h - R_h)| \\
&\overset{(ii)}{\leq} \sum_{k=1}^{d} |\alpha_k(\lambda)| \cdot |\widehat{R}_{h-k} - R_{h-k}| + |\widehat{R}_h - R_h| \\
&\overset{(iii)}{\leq} d \cdot 4^d \cdot \eta + \eta \\
&\leq 2d \cdot 4^d \cdot \eta.
\end{aligned}$$

where the equality (i) follows from the relation (18) and the inequality (ii) follows from Triangle inequality. The inequality (iii) follows by plugging in the bound from Lemma 6 for $|\alpha_k(\lambda)|$ and using the bound in (45). Plugging in the value of $\eta = \min\Big\{\sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}}, \frac{4K^{3d}\log(6d|\Pi|/\delta)}{n}\Big\}$ in the above bound, we get that

$$|\sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda) \widehat{R}_{h-k} - \widehat{R}_h| \leq 2d \cdot 4^d \cdot \min\Big\{\sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}}, \frac{4K^{3d}\log(6d|\Pi|/\delta)}{n}\Big\}. \tag{46}$$

Thus, the vector $\lambda \in \mathbb{C}$ is a feasible solution to the optimization problem in (29). Next, noting the fact that (29) is a minimization problem, we get that for the returned solution $\widehat{\lambda}$ must satisfy

$$\prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\widehat{\lambda}_k|^j\Big) \leq \prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\lambda_k|^j\Big).$$

□

We are now ready to prove our adaptive upper bound in Theorem 3. The proof is very similar to the proof of Theorem 1 given in Appendix C.5. The main technical difference is that we use an adaptive error propagation bound, given in Lemma 13, instead of the error propagation bound from Lemma 3 to control the error in the predicted rewards.

Proof of Theorem 3 . Starting from Lemma 2, we get that with probability at least $1-\delta$, for every policy $\pi \in \Pi$, our estimate $\widehat{R}^{\pi}_{h}$ computed in line 4 of Algorithm 4 satisfies the error bound

$$\max_{h' \le 3d} |\widehat{R}^{\pi}_{h'} - R^{\pi}_{h'}| \le \min\left\{ \sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}}, \frac{4K^{3d}\log(6d|\Pi|/\delta)}{n} \right\}. \tag{47}$$

Now, consider any policy $\pi \in \Pi$, and let $\widehat{\lambda}^{\pi}$, $\widehat{\Delta}^{\pi}$, $\widetilde{R}^{\pi}_{h}$ and $\widetilde{V}^{\pi}$ denote the corresponding local variables in the AdaValEstimate when invoked in Algorithm 4 for the policy π. Further, let λ^{π} denote the eigenvalues of the transition matrix T^{π}. As a consequence of Lemma 1, the expected rewards R^{π}_{h} satisfy an autoregression where the coefficients are determined by λ^{π}. Specifically, for any $h \ge d+1$,

$$R^{\pi}_{h} = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\lambda^{\pi}) \cdot R^{\pi}_{h-k}.$$

Furthermore, by definition (see line 6 of Algorithm 4), the predicted rewards $\widetilde{R}^{\pi}_{h}$ also satisfy a similar autoregression where the coefficients are determined by $\widehat{\lambda}^{\pi}$, the solution of the optimization problem in (29) for the policy π. We have, for any $h \ge d+1$,

$$\widetilde{R}^{\pi}_{h} = \sum_{k=1}^{d} (-1)^{k+1} \alpha_k(\widehat{\lambda}^{\pi}) \cdot \widetilde{R}^{\pi}_{h-k}$$

where $\widetilde{R}_{h'} := \widehat{R}_{h'}$ for $h' \le d$. Additionally, also note that T^{π} is a stochastic matrix and thus $|\lambda^{\pi}_{k}| \le 1$ for all $k \in [d]$. By definition, we also have that $|\widehat{\lambda}^{\pi}_{k}| \le 1$. Thus, using the error propagation bound in Lemma 13 for the sequences $\{R^{\pi}_{h}\}$ and $\{\widetilde{R}^{\pi}_{h}\}$, we get that for any $h \ge 1$,

$$\begin{aligned} |\widetilde{R}^{\pi}_{h} - R^{\pi}_{h}| &\le 4^d h \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\lambda^{\pi}_{k}|^{j} \Big) \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\widehat{\lambda}^{\pi}_{k}|^{j} \Big) \cdot \max_{h' \le 3d} |\widetilde{R}_{h'} - R_{h'}| \\ &\le 4^d h \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\lambda^{\pi}_{k}|^{j} \Big)^{2} \cdot \max_{h' \le 3d} |\widetilde{R}_{h'} - R_{h'}|, \end{aligned} \tag{48}$$

where the inequality in the second line above follows from the fact that

$$\prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\widehat{\lambda}^{\pi}_{k}|^{j} \Big) \le \prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\lambda^{\pi}_{k}|^{j} \Big)$$

as a consequence of Lemma 16-(a) for the policy π. Next, Lemma 16-(b) for the policy π implies that the predicted rewards $\widetilde{R}^{\pi}_{h'}$ satisfy the error bound

$$\begin{aligned} \max_{h' \le 3d} |\widetilde{R}^{\pi}_{h'} - R^{\pi}_{h'}| &\le 2d \cdot (64e)^{d} \cdot \max_{h' \le 3d} |\widehat{R}^{\pi}_{h'} - R^{\pi}_{h'}| \\ &\le 2d \cdot (64e)^{d} \cdot \eta, \end{aligned}$$

where η denotes the right hand side of (47). Plugging the above in (48), we get that

$$|\widetilde{R}^{\pi}_{h} - R^{\pi}_{h}| \le 2dh(256e)^{d} \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\lambda^{\pi}_{k}|^{j} \Big)^{2} \cdot \eta. \tag{49}$$

for any $h \ge 1$. Thus, the error in the estimated value $\widetilde{V}^{\pi}$ for the policy π is bounded by

$$|\widetilde{V}^{\pi} - V^{\pi}| = |\sum_{h=1}^{H} (\widetilde{R}^{\pi}_{h} - R^{\pi}_{h})|$$

$$
\begin{aligned}
&\leq \sum_{h=1}^{H} |\widetilde{R}_h^{\pi} - R_h^{\pi}| \\
&\leq \sum_{h=1}^{H} 2dh(256e)^d \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{h-1} |\lambda_k^{\pi}|^j \Big)^2 \cdot \eta \\
&\leq 2dH^2(256e)^d \cdot \prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\lambda_k^{\pi}|^j \Big)^2 \cdot \eta \\
&\leq 2dH^2(256e)^d \cdot \max_{\pi' \in \Pi} \prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\lambda_k^{\pi'}|^j \Big)^2 \cdot \eta, \qquad (50)
\end{aligned}
$$

where the inequality in the second last line follows by using the bound in (49).

Since π is arbitrary in the above chain of arguments, the error bound in (50) holds for all policies $\pi \in \Pi$. Thus, for any $\pi \in \Pi$, the policy $\widetilde{\pi}$ returned in line 4 of Algorithm 3 satisfies

$$
\begin{aligned}
V^{\widetilde{\pi}} - V^{\pi} &= (\widetilde{V}^{\pi} - V^{\pi}) + (\widetilde{V}^{\widetilde{\pi}} - \widetilde{V}^{\pi}) + (V^{\widetilde{\pi}} - \widetilde{V}^{\widetilde{\pi}}) \\
&\geq (\widetilde{V}^{\pi} - V^{\pi}) + (V^{\widetilde{\pi}} - \widetilde{V}^{\widetilde{\pi}}) \\
&\geq -|\widetilde{V}^{\pi} - V^{\pi}| - |V^{\widetilde{\pi}} - \widetilde{V}^{\widetilde{\pi}}|,
\end{aligned}
$$

where the inequality in the second line follows from the fact that $\widetilde{V}^{\widetilde{\pi}} \geq \widetilde{V}^{\pi}$ for every $\pi \in \Pi$ by the definition of the policy $\widetilde{\pi}$. Using the bound from (50) for policies π and $\widetilde{\pi} \in \Pi$ in the above, we get that

$$
\begin{aligned}
V^{\widetilde{\pi}} &\geq V^{\pi} - 4dH^2(256e)^d \cdot \max_{\pi' \in \Pi} \prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\lambda_k^{\pi'}|^j \Big)^2 \cdot \eta \\
&\geq V^{\pi} - 4dH^2(256e)^d \cdot \max_{\pi' \in \Pi} \prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\lambda_k^{\pi'}|^j \Big)^2 \cdot \min\Big\{ \sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}}, \frac{4K^{3d}\log(6d|\Pi|/\delta)}{n} \Big\} \\
&\geq V^{\pi} - 4dH^2(256e)^d \cdot \max_{\pi' \in \Pi} \prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\lambda_k^{\pi'}|^j \Big)^2 \sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}}
\end{aligned}
$$

where the inequality in the second line above follows by plugging in the value of η as the right hand side of (47), and the inequality in the last line holds due to the fact that $-\min\{a, b\} \geq -a$ for any $a, b \geq 0$.

Since the above holds for any $\pi \in \Pi$, we have that

$$
V^{\widetilde{\pi}} \geq \max_{\pi \in \Pi} V^{\pi} - 4dH^2(256e)^d \cdot \max_{\pi' \in \Pi} \prod_{k=2}^{d} \Big(\sum_{j=0}^{H-1} |\lambda_k^{\pi'}|^j \Big)^2 \sqrt{\frac{8K^{3d}\log(6d|\Pi|/\delta)}{n}},
$$

hence proving the desired statement. □

D.4 Adaptivity to rank

We now describe how the learner can find the best policy in the class Π, that satisfies Assumption 1, without knowing the value of the rank parameter. Let us denote the unknown rank parameter by d^*. Our adaptive algorithm, given in Algorithm 5, follows from standard techniques in the model selection literature. For every $d \in [H]$, we compute an optimal policy $\widetilde{\pi}_d$ assuming that the rank $d^* = d$. Then, for each $d \in [H]$, we estimate the value function for the policy $\widetilde{\pi}_d$ by drawing $n/2H$ fresh trajectories using that policy. Finally, we return the policy $\widetilde{\pi}$ from the set $\{\widetilde{\pi}_d\}_{d \in [H]}$ with the highest estimated value. The returned policy $\widetilde{\pi}$ satisfies, with probability at least $1 - \delta$,

$$
V^{\widetilde{\pi}} \geq \max_{\pi \in \Pi} V^{\pi} - O\Big(\Big(\frac{H}{d^*} \Big)^{2d^*} \sqrt{\frac{(8K)^{3d^*}\log(6d|\Pi|/\delta)}{n}} - 2\sqrt{\frac{\log(H)\log(1/\delta)}{n}} \Big). \qquad (51)
$$

Algorithm 5 Adaptive policy search algorithm (adaptivity to rank)

Input: horizon H, rank d, number of episodes n, finite policy class Π

1: Collect a dataset $\mathcal{D} = \{(x_h^t, a_h^t, r_h^t)_{h=1}^H\}_{t=1}^{n/2}$ by sampling $n/2$ trajectories where actions are sampled from Uniform($\mathcal{A}$).
2: **for** $d \in \{1, 2, \ldots, H\}$ **do**
3: **for** policy $\pi \in \Pi$ **do**
4: Estimate $\widetilde{V}_d^\pi$ by calling ValEstimate($H, d, \mathcal{D}, \pi$).
5: Compute the policy $\widetilde{\pi}_d \in \operatorname{argmax}_{\pi \in \Pi} \widetilde{V}_d^\pi$.
6: Collect $n/2H$ more episodes using the policy $\widetilde{\pi}_d$ and estimate the value $\bar{V}^{\widetilde{\pi}_d}$ using the empirical average of the returned rewards.
7: **Return:** policy $\widetilde{\pi}$ with best estimated value $\widetilde{\pi} \in \operatorname{argmax}_{d \in [H]} \bar{V}^{\widetilde{\pi}_d}$.

Note that, in Algorithm 5, we cap the value of d^* by H. In the case, when $d^* > H$, we can directly estimate the expected reward for each policy by importance sampling upto H steps, and thus compute the optimal policy in Π.

Finally, we can get an algorithm that adapts to both the unknown rank d^* and the eigenspectrum simultaneously by using the procedure AdaValEstimate (given in Algorithm 4) instead of the procedure ValEstimate in Algorithm 5. This implies the following adaptive bound for well mixing MDPs.

Corollary 2 (Well mixing MDP)**.** *Given $\delta \in (0, 1)$, horizon H, a policy class Π and a MDP M.*

(a) If for every policy $\pi \in \Pi$, the transition matrix T^π has at most d^* non-zero eigenvalues such that the second largest eigenvalue $|\lambda_2^\pi| \leq 1 - \gamma$, where K and γ are not known to the learner. Then, Algorithm 5 (run win AdaValEstimate instead of ValEstimate) returns a policy $\widetilde{\pi}$ such that, with probability at least $1 - \delta$,

$$V^{\widetilde{\pi}} \geq \max_{\pi \in \Pi} V^\pi - \widetilde{O}\Big(\Big(\frac{K}{\gamma}\Big)^{2d^*} \frac{1}{\sqrt{n}}\Big).$$

(b) If for every policy $\pi \in \Pi$, the mixing time of the transition matrix T^π is bounded by τ, where τ is not known to the learner. Then, Algorithm 5 (run win AdaValEstimate instead of ValEstimate) returns a policy $\widetilde{\pi}$ such that, with probability at least $1 - \delta$,

$$V^{\widetilde{\pi}} \geq \max_{\pi \in \Pi} V^\pi - \widetilde{O}\Big(\frac{K^{2\tau}}{\sqrt{n}}\Big).$$

The exponential dependence in the mixing time in the above performance guaranee is unavoidable without further assumptions as illustrated by our lower bounds construction in Section 5.

E Lower bounds

E.1 Lower bound construction

We start by describing the lower bound construction, consisting of the policy class Π and the family M of Markov decision processes with rank $2d+2$. All MDPs in the family M have the observation space $\mathcal{X}$ of finite (but very large) size $N = |\mathcal{X}|$ and action space $\mathcal{A} = \{0, 1\}$, but have different transition dynamics.

Policy class Π. The policy class $\Pi \subset \{\mathcal{X} \mapsto \mathcal{A}\}$ consists of $K = (H/d)^d$ deterministic policies that are sufficiently distinct from each other. Specifically, for any two distinct policies $(\pi, \pi') \in \Pi^2$,

$$\sum_{x \in \mathcal{X}} \mathbb{1}\{\pi(x) \neq \pi'(x)\} \geq \frac{N}{4}.$$

Existence of such a policy class follows from the Gilbert-Varshamov bound (Lemma 18) when $8d \log(H/d) \leq N$.

Family of MDPs M. Each MDP $M_{\pi,\phi} \in \mathsf{M}$ is indexed by a policy $\pi \in \Pi$ (which will be optimal in that MDP) and a function $\phi \colon \mathcal{X} \mapsto \mathcal{S}$ that maps each observation in $x \in \mathcal{X}$ to one of the $2d+2$ hidden states given by $S = \{(1, g), (1, b), \ldots, (d, g), (d, b), +, -\}$. In the following, we describe the transition dynamics and the reward function for the MDP $M_{\pi^*,\phi}$.

Transition dynamics of $M_{\pi,\phi}$. The transition dynamics of $M_{\pi,\phi}$ is governed by the mapping ϕ and the dynamics in the $2d+2$ latent states. The dynamics in the latent states S is given by two parallel chains, depicted in Figure 2. Each latent state, except for the final states $+$ and $-$, have the form (i, g) or (i, b) where $i \in [d]$ denotes the index in the chain, and the notation g and b denotes good states and bad states respectively. The initial observation x_0 always corresponds to the hidden state $(1, g)$. At each time step, independent of the action taken, the chain index i increases by 1 with probability p_i (defined later) or stays the same with probability $(1 - p_i)$. As long as the agent follows actions according to π, the next latent state remains a good state (with the second component g). However, as soon as the agent takes an action that π would not have taken, the second component is set to b and then stays b forever. If the agent reaches latent state (d, g) it transitions to the latent state $+$ with probability $\frac{1}{2} + \varepsilon$ and to the latent state $-$ with probability $\frac{1}{2} - \varepsilon$. From (d, b), the agent transitions to both the latent states $+$ or $-$ with equal probability. Finally, from the hidden state $+$, the agent transitions to $-$ in the next step with probability 1. The state $-$ always transitions back to itself independent of the action taken.

We next describe, how the above dynamics in the latent state space defines the transition dynamics for the MDP $M_{\pi,\phi}$ in the observation space. Define $\mathcal{X}_s := \{x \in \mathcal{X} \mid \phi(x) = s\}$ as the set of observations from $\mathcal{X}$ that are mapped to latent state $s \in \mathcal{S}$ by the feature mapping ϕ, and define $D_s := \text{Uniform}(\mathcal{X}_s)$ to denote the uniform distribution over the set $\mathcal{X}_s$. The initial observation x_0 is sampled independently from $\mu_0 = D_{(1,g)}$. The two parameters π and ϕ of MDP $M_{\pi,\phi}$ define the transition dynamics $T_{\pi,\phi}$ as follows:

a) For any observations $x \in \mathcal{X}_{(i,g)}$ of **good latent states**, where $i \in [d-1]$,

$$T_{\pi,\phi}(x, a) = \begin{cases} p_i D_{(i+1,g)} + (1 - p_i) D_{(i,g)} & \text{if } a = \pi(x) \\ p_i D_{(i+1,b)} + (1 - p_i) D_{(i,b)} & \text{else} \end{cases},$$

where the value of $p_i \in (0, 1)$ is set later and $T_{\pi,\phi}(x, a)$ denotes the probability distribution over the next observation x' when taking action a at observation x.

b) For any observations $x \in \mathcal{X}_{(i,b)}$ of **bad latent states**, where $i \in [d-1]$ and all $a \in \mathcal{A}$,

$$T_{\pi,\phi}(x, a) = p_i D_{(i+1,b)} + (1 - p_i) D_{(i,b)}.$$

c) For any observations $x \in \mathcal{X}_{(d,g)}$ of the **good goal state** and all $a \in \mathcal{A}$,

$$T_{\pi,\phi}(x, a) = \left(\frac{1}{2} + \varepsilon\right) D_+ + \left(\frac{1}{2} - \varepsilon\right) D_-,$$

where the bias $\varepsilon \in (0, 1/2)$ is set later.

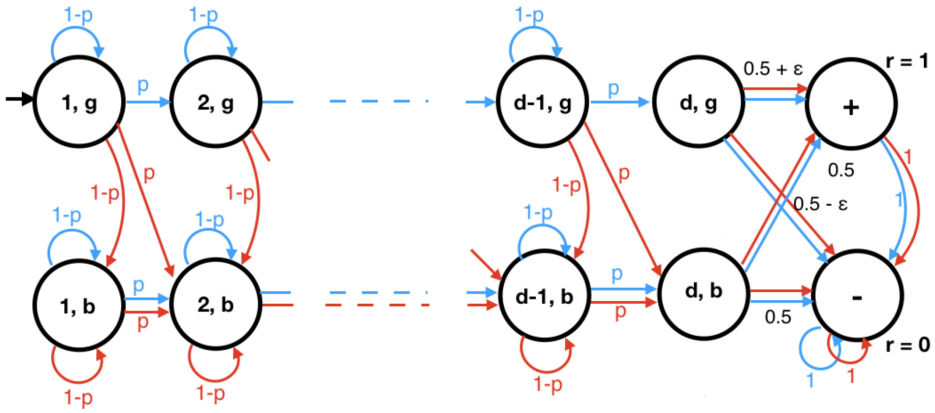

Figure 2: Latent state construction: contextual combination lock. As long as the agent follows actions of the policy π that characterizes the MDP $M_{\phi,\pi}$ (blue arrows), the agent remains in good states (i, g) and receives a Bernoulli$(1/2 + \varepsilon)$ reward but otherwise transits to bad states (i, b) and receives a Bernoulli$(1/2)$ reward.

d) For any observation $x \in \mathcal{X}_{(d,b)}$ of the **bad goal state**, and all $a \in \mathcal{A}$,

$$T_{\pi,\phi}(x, a) = \frac{1}{2}D_+ + \frac{1}{2}D_-.$$

e) For any observation $x \in \mathcal{X}_- \cup \mathcal{X}_+$ of **latent states** $-$ **and** $+$, and all $a \in \mathcal{A}$,

$$T_{\pi,\phi}(x, a) = D_-.$$

Reward function for $M_{\phi,\pi}$. For any observation x, the reward is 0 unless the latent state correspond to x is $+$, in which case, the reward is 1. Specifically,

$$r(x, a) = \mathbb{1}\{x \in \mathcal{X}_+\}\,.$$

Initial observation in $M_{\phi,\pi}$. The initial observation x_0 is sampled uniformly at random from $\mathcal{X}_{(1,g)}$.

Additional MDP $M_{0,\phi}$. In addition to the above defined MDPs $M_{\pi,\phi}$, we define the MDP $M_{0,\phi}$ for every ϕ in the MDP where latent states with b and g behave exactly the same. Specifically, the transition dynamics is given by

$$T_{0,\phi}(x, a) = \begin{cases} \frac{1}{2}D_+ + \frac{1}{2}D_- & \text{if } x \in X_{(d,b)} \cup X_{(d,g)} \\ \frac{p}{2}D_{(i+1,b)} + \frac{1-p}{2}D_{(i,b)} + \frac{p}{2}D_{(i+1,g)} + \frac{1-p}{2}D_{(i,g)} & \text{if } x \in X_{(i,b)} \cup X_{(i,g)} \text{ for } i \in [d-1]\,. \\ D_- & \text{if } x \in X_+ \cup X_- \end{cases}$$

Note that the actions taken do not affect the rewards or observations received in MDPs $M_{0,\phi}$ and thus every policy is an optimal policy.

The family of MDPs M is finally defined as

$$\mathsf{M} := \{M_{\pi,\phi} \mid \pi \in \Pi,\ \phi \in \mathcal{X} \mapsto \mathcal{S}\} \cup \{M_{0,\phi} \mid \phi \in \mathcal{X} \mapsto \mathcal{S}\}.$$

We note that the rank of each MDP in the class M is $O(d)$ as show in the following lemma.

Lemma 17 (Rank bound for MDPs in M). *Let $M_{\pi^*,\phi}$ be an MDP in M. Let $\pi \in \Pi$ be any policy and let $T^{\pi}_{\pi^*,\phi}$ denote the induced transition matrix of the policy π in the MDP $M_{\pi^*,\phi}$. Then, the rank of the matrix $T^{\pi}_{\pi^*,\phi}$ is bounded as*

$$\operatorname{rank}(T^{\pi}_{\pi^*,\phi}) \le 2d - 1 \qquad \textit{and} \qquad \operatorname{rank}(T^{\pi}_{0,\phi}) \le 2d - 1\,.$$

Further, the non-zero eigenvalues of $T^{\pi}_{\pi^*,\phi}$ and $T^{\pi}_{0,\phi}$ are given by

$$\begin{cases} 1-p_i & \textit{for } s=(i,b) \textit{ where } i\in[d-1] \\ (1-p_i)\Pr_{x\sim \textit{Unif}(\mathcal{X}_s)}(\pi^*(x)=\pi(x)) & \textit{for } s=(i,g) \textit{ where } i\in[d-1] \\ 1 & \textit{for } s=-. \end{cases}$$

and

$$\begin{cases} \frac{1-p_i}{2} & \textit{for } s=(i,g) \textit{ where } i\in[d-1] \\ \frac{1-p_i}{2} & \textit{for } s=(i,b) \textit{ where } i\in[d-1] \\ 0 & \textit{for } s\in\{+,(d,g),(d,b))\} \\ 1 & \textit{for } s=-. \end{cases}$$

respectively.

Proof. We can write the transition probability from observation x to x' as

$$\begin{aligned} T^{\pi}_{\pi^*,\phi}(x'|x) =& \mathbb{1}\{\pi(x)=\pi^*(x)\}P_{\text{good}}(\phi(x')|s=\phi(x))\frac{1}{|\mathcal{X}_{\phi(x')}|} \\ &+ \mathbb{1}\{\pi(x)\neq\pi^*(x)\}P_{\text{bad}}(\phi(x')|s=\phi(x))\frac{1}{|\mathcal{X}_{\phi(x')}|} \end{aligned}$$

where $P_{\text{good}}\in\mathbb{R}^{\mathcal{S}\times\mathcal{S}}$ and $P_{\text{bad}}\in\mathbb{R}^{\mathcal{S}\times\mathcal{S}}$ are the latent state transition kernels when the agent follows a good action and bad action respectively. Without loss of generality, we can assume that latent states are ordered as

$$(1,g),(2,g),\ldots,(d,g),(1,b),(2,b),\ldots,(d,b),+,-$$

in which case the agent can only move forward (or stay in the same state) in this order. Thus, when writing P_{good} and P_{bad} as matrices over $\mathcal{S}\times\mathcal{S}$ in this order, they are upper-triangular matrices. Their eigenvalues correspond to the entries on the diagonal and hence, the probability of staying in each latent state is an eigenvalue, including $1-p_i$ for states (i,b) all $i\in[d-1]$ for both P_{good} and P_{bad}. In matrix form, the transition matrix over observations can be written as

$$T^{\pi}_{\pi^*,\phi}=\frac{|\mathcal{S}|}{|\mathcal{X}|}I_{\text{good}}\Phi^T P_{\text{good}}\Phi+\frac{|\mathcal{S}|}{|\mathcal{X}|}I_{\text{bad}}\Phi^T P_{\text{bad}}\Phi,$$

where $\Phi\in\mathbb{R}^{\mathcal{S}\times\mathcal{X}}$ with $\Phi_{s,x}=\mathbb{1}\{\phi(x)=s\}$ is a matrix form of ϕ and $I_{\text{good}}\in\mathbb{R}^{\mathcal{X}\times\mathcal{X}}$ and $I_{\text{bad}}\in\mathbb{R}^{\mathcal{X}\times\mathcal{X}}$ are diagonal matrices with entries $[I_{\text{good}}]_{x,x}=\mathbb{1}\{\pi(x)=\pi^*(x)\}$ and $[I_{\text{bad}}]_{x,x}=\mathbb{1}\{\pi(x)\neq\pi^*(x)\}$, respectively. By the Weinstein–Aronszajn identity, the eigenvalues of $T^{\pi}_{\pi^*,\phi}$ are identical to the eigenvalues of

$$\frac{|\mathcal{S}|}{|\mathcal{X}|}\Phi I_{\text{good}}\Phi^T P_{\text{good}}+\frac{|\mathcal{S}|}{|\mathcal{X}|}\Phi I_{\text{bad}}\Phi^T P_{\text{bad}}=I_{\mathcal{S},\text{good}}P_{\text{good}}+I_{\mathcal{S},\text{bad}}P_{\text{bad}},$$

where $I_{\mathcal{S},\text{good}}\in\mathbb{R}^{\mathcal{S}\times\mathcal{S}}$ and $I_{\mathcal{S},\text{bad}}\in\mathbb{R}^{\mathcal{S}\times\mathcal{S}}$ are diagonal matrices that contain for each $s\in\mathcal{S}$ the probability that policy π matches π^* or does not match π^* on observations of s, respectively. Finally, $I_{\mathcal{S},\text{good}}P_{\text{good}}+I_{\mathcal{S},\text{bad}}P_{\text{bad}}$ is also an upper triangular matrix whose eigenvalues are the entries on the diagonal. Therefore, the eigenvalues of this matrix and $T^{\pi}_{\pi^*,\phi}$ are

$$\begin{cases} 1-p_i & \text{for } s=(i,b) \text{ where } i\in[d-1] \\ (1-p_i)\Pr_{x\sim\text{Unif}(\mathcal{X}_s)}(\pi^*(x)=\pi(x)) & \text{for } s=(i,g) \text{ where } i\in[d-1] \\ 0 & \text{for } s\in\{+,(d,g),(d,b))\} \\ 1 & \text{for } s=-. \end{cases}$$

Thus, the rank of $T^{\pi}_{\pi^*,\phi}$ is at most $2d-1$. Analogously, we can show that the eigenvalues of $T^{\pi}_{0,\phi}$ are

$$\begin{cases} \frac{1-p_i}{2} & \text{for } s=(i,b) \text{ and } (i,g) \text{ where } i\in[d-1] \\ 0 & \text{for } s\in\{+,(d,g),(d,b))\} \\ 1 & \text{for } s=-. \end{cases}$$

Thus, the rank of $T^{\pi}_{\pi^*,\phi}$ is at most $2d-1$. □

Lemma 18 (Gilbert-Varshamov bound [Massart, 2007])**.** *Let $N > 1$. There exists a subset $\mathcal{V}$ of $\{0,1\}^N$ of size $|\mathcal{V}| \geq \exp(N/8)$ such that*

$$\sum_{i=1}^{N} \mathbb{1}\{v_i \neq v_i'\} \geq \frac{N}{4} \tag{52}$$

for all $v, v' \in \mathcal{V}$.

E.2 Proof of Theorem 2

In the following, we provide the lower bound which states that the factor of $\Omega(H^d)$ in unavoidable without making further assumptions. We restate Theorem 2 here with explicit constants:

Theorem 7. *Let $\widetilde{\varepsilon} \in (0, 1/26)$, $\delta \in (0, 1/2)$, $d \geq 4$ and $H \geq 219d$. There exists a realizable policy class of size $(H/d)^d$ and a family of MDPs with rank at most $2d$, finite observation space, horizon H and two actions such that: Any algorithm that returns an $\widetilde{\varepsilon}$-optimal policy in any MDP in this family with probability at least $1 - \delta$ has to collect at least*

$$\frac{1}{12168 \cdot H\widetilde{\varepsilon}^2}\left(\frac{H}{41d}\right)^{d/2} \log\left(\frac{1}{2\delta}\right)$$

episodes in expectation in some MDP in this family.

Proof. Consider any $(\widetilde{\varepsilon}, \delta)$-PAC RL algorithm A . Our lower bound is based on the policy class Π and the family of MDPs M constructed in Appendix E.1. We set $|\Pi| = (H/d)^d$ and $p_i = d/H$ for all $i \in [d]$.

We first define additional notation. Let the random variable G denote the first time-step in an episode when an observation from the latent state (d, g) or (d, g) is observed. In order to reach these latent states, the agent is required to do $d - 1$ latent state progressions, each happening with probability $p_1, \ldots, p_{d-1}$ respectively. Furthermore, for our constructions in Appendix E.1 of the class $\mathcal{M}$, we note that the distribution of G only depends on $\{p_i\}_{i<d}$, d and H, but is otherwise independent of the MDP instance, the parameter ε and the played policy. In fact, when $p_i = d/H$, an application of Lemma 21 implies that

$$\Pr(G \leq H - 1) \geq 1 - \exp(-2/5). \tag{53}$$

Now, consider any MDP $M_{\pi^*,\phi} \in \mathsf{M}$ and let $V_{\pi^*,\phi}(\pi)$ denote the expected return of the policy π in the MDP $M_{\pi^*,\phi}$. From our MDP construction, we note that for the optimal policy π^*,

$$V_{\pi^*,\phi}(\pi^*) = \left(\frac{1}{2} + \varepsilon\right) \Pr(G \leq H - 1).$$

Similarly, for any other policy $\pi \in \Pi$, we have[1]

$$V_{\pi^*,\phi}(\pi) = \frac{1}{2} \Pr(G \leq H - 1) + \varepsilon \mathbb{E}^{\pi}_{\pi^*,\phi}\left[\mathbb{1}\{\pi(X_{1:G-1}) = \pi^*(X_{1:G-1})\}\right] \Pr(G \leq H - 1),$$

where $\{\pi(X_{1:G-1}) = \pi^*(X_{1:G-1})\}$ denotes the event that the action chosen by π agrees with that chosen by π^* on the observations $X_{1:G-1}$ up to time step $G - 1$. Hence, we have that the suboptimality gap for the policy π is

$$V_{\pi^*,\phi}(\pi^*) - V_{\pi^*,\phi}(\pi) = \varepsilon \Pr(G \leq H - 1) \Pr^{\pi}_{\pi^*,\phi}(\exists h \leq G - 1 \text{ s.t. } \pi(X_h) \neq \pi^*(X_h)) \,. \tag{54}$$

Next, define the random variable τ to denote the number of episodes after which the algorithm A terminates and let $\widehat{\pi}$ denote the policy returned on termination. Both, τ and $\widehat{\pi}$, depend on the algorithm A and the underlying MDP on which A collects data from. Since the algorithm A is $(\widetilde{\varepsilon}, \delta)$-PAC, we have that for any MDP $M_{\pi^*,\phi}$, with probability at least $1 - \delta$,

$$V_{\pi^*,\phi}(\pi^*) - V_{\pi^*,\phi}(\widehat{\pi}) \leq \widetilde{\varepsilon}.$$

[1] Throughout the proof, for any random variable Y, we define the notation $\mathbb{E}^{\pi}_{\pi^*,\phi}[Y]$ to denote the expectation of Y where the trajectory is drawn using the policy π in the MDP $M_{\pi^*,\phi}$.

Using the relation in (54), and plugging in the bound in (53), in the above, we get that

$$\varepsilon \Pr(G \le H-1) \Pr^{\widehat{\pi}}_{\pi^*,\phi}(\exists h \le G-1 \ \text{ s.t. } \widehat{\pi}(X_h) \ne \pi^*(X_h)) \le \widetilde{\varepsilon}, \tag{55}$$

must hold with probability at least $1-\delta$ for any MDP $M_{\pi^*,\phi}$. For our lower bound constructions, we set

$$\varepsilon = \frac{4\widetilde{\varepsilon}}{\Pr(G \le H-1)} \le 13\widetilde{\varepsilon}, \tag{56}$$

and thus (55) implies that $\Pr^{\widehat{\pi}}_{\pi^*,\phi}(\widehat{\pi}(X_{1:G-1}) = \pi^*(X_{1:G-1})) \ge 3/4$ must hold with probability at least $1-\delta$. Define the event

$$\mathrm{Opt}^{\mathsf{A}}_{\pi^*,\phi} := \left\{\Pr^{\widehat{\pi}}_{\pi^*,\phi}(A_{1:G-1} = \pi^*(X_{1:G-1})) \ge 3/4\right\}.$$

The above analysis suggests that for any $M_{\pi^*,\phi}$

$$\Pr_{\pi^*,\phi}\left(\mathrm{Opt}^{\mathsf{A}}_{\pi^*,\phi}\right) \ge 1-\delta. \tag{57}$$

Next, for any $\pi^* \in \Pi$, define the measure $\Pr_{\pi^*}(Y) = \frac{1}{|\Phi|}\sum_{\phi\in\Phi}\Pr_{\pi^*,\phi}(Y)$, i.e. the probability measure induced by first picking ϕ uniformly at random from the set of all mappings Φ and then considering the distribution induced by $M_{\pi^*,\phi}$. The measure $\Pr_0(Y) = \frac{1}{|\Phi|}\sum_{\phi\in\Phi}\Pr_{0,\phi}(Y)$ is defined analogously for the MDP $M_{0,\phi}$. Thus, from (57), we have that for any $\pi^* \in \Pi$,

$$\Pr_{\pi^*}(\mathrm{Opt}^{\mathsf{A}}_{\pi^*,\phi}) \ge 1-\delta. \tag{58}$$

We are now ready to prove the desired lower bound. Let

$$T_{\max} := \frac{1}{\delta}\cdot\frac{1}{12168H\widetilde{\varepsilon}^2}\left(\frac{H}{41d}\right)^{d/2}\cdot\log(1/2\delta).$$

There are two natural scenarios: either (a) $\Pr_{\pi^*}(\tau > T_{\max}) > \delta$ for some $\pi^* \in \Pi$, or (b) $\Pr_{\pi^*}(\tau > T_{\max}) \le \delta$ for all $\pi^* \in \Pi$. We analyse the two cases separately below.

Case-(a): $\Pr_{\pi^*}(\tau > T_{\max}) > \delta$ for some $\pi^* \in \Pi$. The lower bound follows immediately in this case. Note that,

$$\max_{\phi\in\Phi}\mathbb{E}_{\pi^*,\phi}[\tau] > \mathbb{E}_{\pi^*}[\tau] \ge \Pr_{\pi^*}(\tau > T_{\max})\cdot T_{\max} \ge \delta T_{\max}\,.$$

Hence, there exists an MDP in $M_{\pi^*,\phi} \in \mathsf{M}$ for which the expected number of episodes collected by the algorithm A is at least $\delta T_{\max}$, which is the desired lower bound.

Case-(b): $\Pr_{\pi^*}(\tau > T_{\max}) \le \delta$ for all $\pi^* \in \Pi$. Due to (58), for any policy $\pi^* \in \Pi$, we have:

$$\begin{aligned}\Pr_{\pi^*}(\tau \le T_{\max} \wedge \mathrm{Opt}^{\mathsf{A}}_{\pi^*,\phi}) &= \Pr_{\pi^*}(\mathrm{Opt}^{\mathsf{A}}_{\pi^*,\phi}) - \Pr_{\pi^*}(\tau > T_{\max} \wedge \mathrm{Opt}^{\mathsf{A}}_{\pi^*,\phi}) \\ &\ge 1-2\delta.\end{aligned}$$

The above condition intuitively states that the policy returned by the algorithm will, with high probability, match the actions of the optimal policy for $G-1$ time steps for any policy $\pi^* \in \Pi$. On the other hand, we show in Lemma 25 through a packing argument that the expected number of policies that can be matched for $G-1$ steps when observations are drawn uniformly is bounded, i.e.

$$\mathbb{E}_{unif}\big[\sum_{\pi^*}\mathbb{1}\{\pi^*(X_{1:G}) = \pi(X_{1:G})\}\big] \le (41\log(H/d))^d H + 2\,,$$

where the notation $\mathbb{E}_{unif}[\cdot]$ denotes that $X_{1:G}$ are drawn independently from uniform$(\mathcal{X})$. We denote this bound by $C = (41\log(H/d))^d H + 2$. We show in Lemma 19 through a careful information-theoretic argument that the expected stopping time of the algorithm $\mathcal{A}$ on instances $M_{0,\phi}$ is bounded from below as

$$\mathbb{E}_0[\tau] \ge \frac{1}{8\varepsilon^2}\left(\frac{|\Pi|}{C} - \frac{8}{3}\right)\log(1/2\delta) - \left(2T_{\max} + \frac{7}{12\varepsilon^2}\log(1/2\delta)\right)\cdot\frac{|\Pi|}{C}\cdot\Delta(T_{\max})\,,$$

where $\Delta(T_{\max}) := 4T_{\max}^2 H^2 |\mathcal{S}|/N$ accounts for the differences in observation distributions in different instances of M.

Plugging in the value of $|\Pi|$ and C, we note that

$$\frac{|\Pi|}{C} \geq \frac{(H/d)^d}{2H(41\log(H/d))^d} = \frac{1}{2H}\Big(\frac{H}{41d\log(H/d)}\Big)^d \geq \frac{1}{2H}\Big(\frac{H}{41d}\Big)^{d/2}.$$

Additionally, for $d \geq 4$ and $H/d \geq 219$, we have $8/3 \leq (H/41d)^{d/2}/4H$. Combining these bounds yields

$$\mathbb{E}_0[\tau] \geq \frac{1}{6084H\widetilde{\varepsilon}^2}\Big(\frac{H}{41d}\Big)^{d/2}\log(1/2\delta) - \Big(2T_{\max} + \frac{7}{12\varepsilon^2}\log(1/2\delta)\Big)\cdot\frac{|\Pi|}{C}\cdot\Delta(T_{\max}),$$

Finally this bound only depends on the number of observations N through $\Delta(T_{\max})$ which goes to zero as $N \to \infty$. Therefore, we can pick N large enough such that the second term becomes small enough and thus

$$\mathbb{E}_0[\tau] \geq \frac{1}{12168H\widetilde{\varepsilon}^2}\Big(\frac{H}{41d}\Big)^{d/2}\log(1/2\delta).$$

Since this bound holds on average over all MDPs $M_{0,\phi} \in \mathcal{M}$, this lower bound must also hold in at least one specific $M_{0,\phi} \in \mathcal{M}$. This gives us the desired statement. □

For the rest of the section, we will build on the notation introduced in the above proof. The following technical lemma gives a lower bound on $\mathbb{E}_0[\tau]$ for the case-(b) above.

Lemma 19. *Let A be any $(\widetilde{\varepsilon},\delta)$-PAC RL algorithm. Let $T_{\max} \in \mathbb{N}$ and assume that $\Pr_{\pi^*}(\tau \leq T_{\max} \wedge \mathrm{Opt}^{\mathsf{A}}_{\pi^*,\phi}) \geq 1-2\delta$ holds for all $\pi^* \in \Pi$. Further, let $C > 0$ denote an upper-bound on the number of policy matches per episode, i.e., for all $\pi \in \Pi$,*

$$\mathbb{E}_{\mathit{unif}}\Big[\sum_{\pi^*}\mathbb{1}\{\pi^*(X_{1:G}) = \pi(X_{1:G})\}\Big] \leq C.$$

Then the expected stopping time τ for the algorithm A over MDP instances $M_{0,\phi}$ where ϕ is drawn randomly from Φ is bounded from below as

$$\mathbb{E}_0[\tau] \geq \frac{1}{8\varepsilon^2}\Big(\frac{|\Pi|}{C} - \frac{8}{3}\Big)\log(1/2\delta) - \Big(2T_{\max} + \frac{7}{12\varepsilon^2}\log(1/2\delta)\Big)\cdot\frac{|\Pi|}{C}\cdot\Delta(T_{\max}),$$

where $\Delta(T_{\max}) = 4T_{\max}^2 H^2|\mathcal{S}|/N$.

Proof. Let G_i denote the first timestep when the agent reaches the latent state (d,g) or (d,b) in the ith episode collected by the algorithm A. We denote by

$$N_{\pi^*}^{\tau\wedge T_{\max}} = \sum_{i=1}^{\tau\wedge T_{\max}}\mathbb{1}\{A_{i,1:G_i-1} = \pi^*(X_{i,1:G_i-1})\}$$

the number of episodes among the first $\tau \wedge T_{\max} = \min\{\tau, T_{\max}\}$ episodes where the actions $A_{i,1:G_i-1}$ played by A in the ith episode matches those of π^* on the corresponding observations, until the latent state (d,g) or (d,b) was reached. We first lower-bound the expected value of $N_{\pi^*}^{\tau\wedge T_{\max}}$ under the measure induced by $\Pr_0$. To that end, we introduce auxiliary MDPs $M_{0,\pi^*,\phi}$ that are identical to $M_{\pi^*,\phi}$ on all latent states except for (d,g). In $M_{0,\pi^*,\phi}$, we transition to both $+$ and $-$ with equal probability from the latent state (d,g). [2] Analogous to $\Pr_{\pi^*}$, we define $\Pr_{0,\pi^*}$ to denote the law when ϕ is drawn uniformly from Φ beforehand and the underlying MDPS is $M_{0,\pi^*,\phi}$. We also define $\Pr_0$ as the law when π^* is additionally drawn uniformly at random from Π beforehand. Finally, $\mathbb{E}_{0,\pi^*}[\cdot]$ and $\mathbb{E}_0[\cdot]$ are defined as the expectations under Π_{0,π^*} and $\Pr_0$ respectively. Following the standard machinery for lower-bounds [Garivier et al., 2019, Domingues et al., 2021], we get that

$$\mathbb{E}_0[N_{\pi^*}^{\tau\wedge T_{\max}}] \overset{(i)}{\geq} \mathbb{E}_{0,\pi^*}[N_{\pi^*}^{\tau\wedge T_{\max}}] - T_{\max}\Delta(T_{\max})$$

[2] Note that the MDPs $M_{0,\pi^*,\phi}$ are only an analytical tool and do not belong to the class M.

$$\geq \mathbb{E}_{0,\pi^*}[N_{\pi^*}^{\tau\wedge T_{\max}}] \cdot \frac{\mathrm{kl}(1/2, 1/2+\varepsilon)}{4\varepsilon^2} - T_{\max}\Delta(T_{\max})$$
$$= \frac{1}{4\varepsilon^2}\,\mathrm{KL}\Big(\Pr_{0,\pi^*}^{\mathcal{F}_{\tau\wedge T_{\max}}}, \Pr_{\pi^*}^{\mathcal{F}_{\tau\wedge T_{\max}}}\Big) - T_{\max}\Delta(T_{\max}),$$

where the inequality (i) follow from an application of Lemma 24. In the above, for any distributions P and Q, the notation $\mathrm{KL}(P\|Q)$ denotes the KL-divergence between P and Q, and the superscript $\mathcal{F}_{\tau\wedge T_{\max}}$ denotes the conditioning w.r.t. the natural filtration generated by the first $\tau\wedge T_{\max}$ episodes. Further, define $\mathrm{kl}(p,q)$ to denote the KL-divergence of two Bernoulli random variables with means p and q respectively. We now apply Lemma 1 of Garivier et al. [2019] which gives that for any $\mathcal{F}_{\tau\wedge T_{\max}}$-measurable variable random variable Z with values in $[0,1]$, we have that

$$\begin{aligned}
\mathbb{E}_0[N_{\pi^*}^{\tau\wedge T_{\max}}] &\geq \frac{1}{4\varepsilon^2}\,\mathrm{kl}(\mathbb{E}_{0,\pi^*}[Z], \mathbb{E}_{\pi^*}[Z]) - T_{\max}\Delta(T_{\max}) \\
&\overset{(ii)}{\geq} \frac{1}{4\varepsilon^2}(1-\mathbb{E}_{0,\pi^*}[Z])\log\Big(\frac{1}{1-\mathbb{E}_{\pi^*}[Z]}\Big) - \frac{1}{4\varepsilon^2}\log(2) - T_{\max}\Delta(T_{\max}) \\
&\overset{(iii)}{\geq} \frac{1}{4\varepsilon^2}(1-\mathbb{E}_0[Z])\log\Big(\frac{1}{1-\mathbb{E}_{\pi^*}[Z]}\Big) - \frac{\log(2)}{4\varepsilon^2} \\
&\qquad - \Big(T_{\max} + \frac{1}{4\varepsilon^2}\log\Big(\frac{1}{1-\mathbb{E}_{\pi^*}[Z]}\Big)\Big)\Delta(T_{\max}) \qquad (59)
\end{aligned}$$

where the inequality (ii) follows due to the fact that $\mathrm{kl}(p,q) \geq (1-p)\log(1/(1-q)) - \log(2)$, and the inequality (iii) holds from an application of Lemma 24. Next, define the random variable $Z_{\pi*}$ as

$$Z_{\pi^*} = \Pr_{\pi^*}\big(\tau \leq T_{\max} \wedge \mathrm{Opt}_{\pi^*,\phi}^{\mathcal{A}} \mid \mathcal{F}_{\tau\wedge T_{\max}}\big)$$

and note that Z_{π^*} is $\mathcal{F}_{\tau\wedge T_{\max}}$-measurable by construction. Thus, plugging $Z = Z_{\pi^*}$ in (59) and using the fact that $\mathbb{E}_{\pi^*}[Z_{\pi^*}] = \Pr_{\pi^*}(\tau \leq T_{\max} \wedge \mathrm{Opt}_{\pi^*,\phi}^{\mathcal{A}}) \geq 1-2\delta$ (by assumption), we get that

$$\mathbb{E}_0[N_{\pi^*}^{\tau\wedge T_{\max}}] \geq \frac{1}{4\varepsilon^2}(1-\mathbb{E}_0[Z_{\pi^*}])\log(1/2\delta) - \frac{\log(2)}{4\varepsilon^2} - \Big(T_{\max} + \frac{1}{4\varepsilon^2}\log(1/2\delta)\Big)\Delta(T_{\max}),$$

Summing the above for all policies $\pi^* \in \Pi$ yields that

$$\begin{aligned}
\sum_{\pi^*\in\Pi}\mathbb{E}_0[N_{\pi^*}^{\tau\wedge T_{\max}}] &\geq \frac{1}{4\varepsilon^2}\Big(|\Pi| - \sum_{\pi^*\in\Pi}\mathbb{E}_0[Z_{\pi^*}]\Big)\log(1/2\delta) \\
&\qquad - \frac{|\Pi|\log(2)}{4\varepsilon^2} - \Big(T_{\max} + \frac{\log(1/2\delta)}{4\varepsilon^2}\Big)|\Pi|\Delta(T_{\max}). \qquad (60)
\end{aligned}$$

We further lower bound the above by deriving an upper bound on $\sum_{\pi^*\in\Pi}\mathbb{E}_0[Z_{\pi^*}]$. Note that for any $\pi^* \in \Pi$,

$$\begin{aligned}
Z_{\pi^*} &= \Pr_{\pi^*}\Big(\tau \leq T_{\max} \wedge \mathrm{Opt}_{\pi^*,\phi}^{\mathcal{A}} \mid \mathcal{F}_{\tau\wedge T_{\max}}\Big) \\
&\overset{(i)}{=} \mathbb{E}_{\pi^*}\Big[\mathbb{1}\{\tau \leq T_{\max}\}\mathbb{1}\Big\{\Pr_{\pi^*,\phi}^{\hat{\pi}}(\hat{\pi}(X_{1:G-1}) = \pi^*(X_{1:G-1})) \geq 3/4\Big\} \mid \mathcal{F}_{\tau\wedge T_{\max}}\Big] \\
&\overset{(ii)}{\leq} \frac{4}{3}\mathbb{E}_{\pi^*}\Big[\mathbb{1}\{\tau \leq T_{\max}\}\Pr_{\pi^*,\phi}^{\hat{\pi}}(\hat{\pi}(X_{1:G-1}) = \pi^*(X_{1:G-1})) \mid \mathcal{F}_{\tau\wedge T_{\max}}\Big]
\end{aligned}$$

where the equality (i) above follows from the definition of $\mathrm{Opt}_{\pi^*,\phi}^{\mathcal{A}}$, and the inequality in (ii) holds from an application of Markov's inequality and using the fact that $\mathbb{1}\{\tau \leq T_{\max}\}$ is $\mathcal{F}_{\tau\wedge T_{\max}}$-measurable (by construction). Note that when $\tau \leq T_{\max}$ (the only outcomes where the random variable inside the expectation can be non-zero), we also have that $\hat{\pi}$ is $\mathcal{F}_{\tau\wedge T_{\max}}$-measurable.[3] Thus, the only randomness in $\Pr_{\pi^*,\phi}^{\hat{\pi}}$ above is due to ϕ which affects the distribution of the observations $X_{1:G-1}$ inside $\Pr_{\pi^*,\phi}^{\hat{\pi}}$. However, note that this distribution is exactly the distribution of observations

[3]This assumes a deterministic algorithm but we can handle stochastic algorithms by simply conditioning on $\hat{\pi}$ (and therefore the internal randomness of the algorithm) as well.

in the $(\tau+1)$th episode if we assume (without loss of generality) that the algorithm plays $\hat{\pi}$ in that episode. Thus, we can write the right hand side in the above as

$$Z_{\pi^*} \leq \frac{4}{3}\mathbb{1}\{\tau \leq T_{\max}\}\Pr_{\pi^*}\left(\pi^*(X_{\tau+1,1:G_{\tau+1}-1}) = \hat{\pi}(X_{\tau+1,1:G_{\tau+1}-1}) \mid \mathcal{F}_{\tau\wedge T_{\max}}\right).$$

An application of Lemma 23 in the above implies that

$$Z_{\pi^*} \leq \frac{4}{3}\mathbb{1}\{\tau \leq T_{\max}\}\Big(\Pr_{\text{unif}}(\pi^*(X_{1:G-1}) = \hat{\pi}(X_{1:G-1})) + \frac{2|\mathcal{S}|H^2(T_{\max}+1)}{N}\Big), \tag{61}$$

which further implies that

$$\begin{aligned}\sum_{\pi^*\in\Pi}\mathbb{E}_0[Z_{\pi^*}] &\leq \frac{4}{3}\mathbb{E}_0\Big[\sum_{\pi^*\in\Pi}\Pr_{\text{unif}}(\pi^*(X_{1:G-1}) = \hat{\pi}(X_{1:G-1}))\Big] + \frac{4}{3}|\Pi|\Delta(T_{\max}) \\ &= \frac{4}{3}C + \frac{4}{3}|\Pi|\Delta(T_{\max}),\end{aligned} \tag{62}$$

where the value of C and $\Delta(T_{\max})$ are given in the lemma statement. Plugging the above bound in (60), we get that

$$\sum_{\pi^*\in\Pi}\mathbb{E}_0[N_{\pi^*}^{\tau\wedge T_{\max}}] \geq \frac{1}{4\varepsilon^2}\Big(|\Pi| - \frac{4}{3}C\Big)\log(1/2\delta) - \frac{|\Pi|\log(2)}{4\varepsilon^2} - \Big(T_{\max} + \frac{7\log(1/2\delta)}{12\varepsilon^2}\Big)|\Pi|\Delta(T_{\max}). \tag{63}$$

Relating policy matches to stopping time: In the following, we show an upper bound on $\sum_{\pi^*\in\Pi}\mathbb{E}_0[N_{\pi^*}^{\tau\wedge T_{\max}}]$ that, when taken together with the above lower bound, gives us the desired lower bound on $\mathbb{E}_0[\tau]$. We note that

$$\begin{aligned}\sum_{\pi^*\in\Pi}\mathbb{E}_0[N_{\pi^*}^{\tau\wedge T_{\max}}] &= \sum_{t=1}^{T_{\max}}\mathbb{E}_0\Big[\mathbb{1}\{\tau > t-1\}\sum_{\pi^*\in\Pi}\mathbb{1}\{A_{t,1:G_t-1} = \pi^*(A_{t,1:G_t-1})\}\Big] \\ &= \sum_{t=1}^{T_{\max}}\mathbb{E}_0\Big[\mathbb{1}\{\tau > t-1\}\mathbb{E}_0\Big[\sum_{\pi^*\in\Pi}\mathbb{1}\{\pi_t(X_{t,1:G_t-1}) = \pi^*(X_{t,1:G_t-1})\} \mid \pi_t, \mathcal{F}_{t-1}\Big]\Big] \\ &\overset{(i)}{\leq} \sum_{t=1}^{T_{\max}}\mathbb{E}_0\Big[\mathbb{1}\{\tau > t-1\}\mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*\in\Pi}\mathbb{1}\{\pi_t(X_{1:G-1}) = \pi^*(X_{1:G-1})\}\Big]\Big] \\ &\qquad + T_{\max}|\Pi|\Delta(T_{\max}) \\ &\overset{(ii)}{\leq} \sum_{t=1}^{T_{\max}}\mathbb{E}_0[\mathbb{1}\{\tau > t-1\}C] + T_{\max}|\Pi|\Delta(T_{\max}) \\ &\leq C\mathbb{E}_0[\tau\wedge T_{\max}] + T_{\max}|\Pi|\Delta(T_{\max}) \\ &\leq C\mathbb{E}_0[\tau] + T_{\max}|\Pi|\Delta(T_{\max})\end{aligned} \tag{64}$$

where the inequality (i) follows from an application of Lemma 23 and the inequality (ii) follows from the definition of C given in the lemma statement.

Combining the lower bound in (63) with the upper bound in (64) and rearranging the terms yields that

$$\begin{aligned}\mathbb{E}_0[\tau] &\geq \frac{1}{4\varepsilon^2}\Big(\frac{|\Pi|}{C} - \frac{4}{3}\Big)\log(1/2\delta) - \frac{|\Pi|}{C}\frac{1}{4\varepsilon^2}\log(2) - \Big(2T_{\max} + \frac{7}{12\varepsilon^2}\log(1/2\delta)\Big)\frac{|\Pi|}{C}\Delta(T_{\max}) \\ &\geq \frac{1}{8\varepsilon^2}\Big(\frac{|\Pi|}{C} - \frac{8}{3}\Big)\log(1/2\delta) - \Big(2T_{\max} + \frac{7}{12\varepsilon^2}\log(1/2\delta)\Big)\frac{|\Pi|}{C}\Delta(T_{\max})\end{aligned}$$

where the last inequality is due to the fact that $\delta \leq 1 \leq \exp(2)/4$ and thus $\log(1/2\delta) \geq 2\log(2)$. This concludes the desired statement. □

E.3 Proof of Theorem 4 (eigenspectrum dependent lower bounds)

We here restate Theorem 4 with explicit constants:

Theorem 8 (Adaptive lower bound). *Let $\widetilde{\varepsilon} \in (0, \frac{1}{16})$, $\delta \in (0, \frac{1}{2})$, $d \geq 4$ and $(\lambda_i)_{i\in[d-1]} \in [0,1]^{d-1}$ satisfy*

$$\Big(\frac{16}{3}(15(d-1))^{d-1}\Big)^2 \leq \prod_{i=1}^{d-1} \frac{1}{1-\lambda_i} \leq \frac{8}{7}\exp(H/2) \qquad \text{and} \qquad \sum_{i=1}^{d-1} \frac{1}{1-\lambda_i} \leq \frac{H}{4\ln(4d)} \,.$$

Then, there is a realizable policy class and family of MDPs with rank at most $\Theta(d)$, finite observation space, horizon H and two actions such that: For each $i \in [d]$, policy π and MDP M in this class, there is an eigenvalue of the induced transition matrix T^π_M in $[\lambda_i/2, \lambda_i]$. Furthermore, any algorithm that returns, with probability at least $1-\delta$ an ε-optimal policy for any MDP in this family, has to collect at least

$$\frac{1}{1100\widetilde{\varepsilon}^2}\Big(\frac{1}{15(d-1)}\Big)^{d-1}\sqrt{\prod_{i=1}^{d-1}\frac{1}{1-\lambda_i}}\log(1/2\delta)$$

episodes in expectation in some MDP in this family.

Proof. This theorem follows immediately from setting $p_i = 1-\lambda_i$ with Lemma 10 and Lemma 20 below. □

Lemma 20. *Let $\widetilde{\varepsilon} \in (0, 1/16)$ and let $\mathcal{M}$ be the family of MDPs defined in the proof of Theorem 2 but where the probability p for progression in latent states (i, g) and (i, b) is set to $p_i \in (0, 1)$. If*

$$\Big(\frac{16}{3}(15(d-1))^{d-1}\Big)^2 \leq \prod_{i=1}^{d-1} \frac{1}{p_i} \leq \frac{8}{7}\exp(H/2) \qquad \text{and} \qquad \sum_{i=1}^{d-1} \frac{1}{p_i} \leq \frac{H}{4\ln(4d)} \,.$$

then any learner that returns an $\widetilde{\varepsilon}$-optimal policy in every MDP in this class with probability at least $1-\delta$ has to collect at least

$$\frac{1}{1100\widetilde{\varepsilon}^2}\Big(\frac{1}{15(d-1)}\Big)^{d-1}\sqrt{\prod_{i=1}^{d-1}\frac{1}{p_i}}\log(1/2\delta)$$

episodes in expectation in at least one MDP in the family.

Proof. We follow the proof of Theorem 2 but set

$$T_{\max} := \frac{1}{\delta}\cdot\frac{1}{1100\widetilde{\varepsilon}^2}\Big(\frac{1}{15(d-1)}\Big)^{d-1}\sqrt{\prod_{i=1}^{d-1}\frac{1}{p_i}}\log(1/2\delta)\,.$$

Then one of two cases can happen: Either there is an MDP $M \in \mathcal{M}$ in the class where the algorithm samples in expectation at least $\mathbb{E}_M[\tau] \geq \delta T_{\max}$ episodes, or

$$\Pr_{\pi^*}(\tau \leq T_{\max} \wedge \mathrm{Opt}^{\mathcal{A}}_{\pi^*,\phi}) \geq 1-2\delta$$

holds for all $\pi^* \in \Pi$ where $\mathrm{Opt}^{\mathcal{A}}_{\pi^*,\phi} = \{\Pr^{\hat{\pi}}_{\pi^*,\phi}(A_{1:G-1} = \pi^*(X_{1:G-1})) \geq 3/4\}$ denote this event, where the policy returned by the algorithm $\hat{\pi}$ is $\widetilde{\varepsilon}$-optimal in $M_{\pi^*,\phi}$. Since the first case immediately gives us the desired lower bound, in the following, we consider the case that $\Pr_{\pi^*}(\tau \leq T_{\max} \wedge \mathrm{Opt}^{\mathcal{A}}_{\pi^*,\phi}) \geq 1-2\delta$ holds for every policy $\pi^* \in \Pi$.

An application of Lemma 26 gives us

$$\mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*}\mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\}\Big] \leq C\,,$$

for any policy π with $C = 2 + |\Pi| \cdot \prod_{i=1}^{d-1} \left(p_i \frac{\log |\Pi|}{\log(8/7)} \right)$ as long as

$$|\Pi| \leq \frac{8}{7} \exp(H/2).$$

Applying Lemma 19, the expected stopping time τ of algorithm A on instances $M_{0,\phi}$ is bounded from below as

$$\mathbb{E}_0[\tau] \geq \frac{1}{8\varepsilon^2} \left(\frac{|\Pi|}{C} - \frac{8}{3} \right) \log(1/2\delta) - \left(2T_{\max} + \frac{7}{12\varepsilon^2} \log(1/2\delta) \right) \frac{|\Pi|}{C} \Delta(T_{\max}) \,.$$

We now set $|\Pi| = \prod_{i=1}^{d-1} \frac{1}{p_i}$ and bound the ratio

$$\begin{aligned}
\frac{|\Pi|}{C} &\geq \min\left\{ \frac{1}{4} \prod_{i=1}^{d-1} \frac{1}{p_i}, \frac{\prod_{i=1}^{d-1} \frac{1}{p_i}}{\left(\ln \prod_{i=1}^{d-1} \frac{1}{p_i} \right)^{d-1}} \left(\ln \frac{8}{7} \right)^{d-1} \right\} \\
&\geq \min\left\{ \frac{1}{4} \prod_{i=1}^{d-1} \frac{1}{p_i}, \sqrt{\prod_{i=1}^{d-1} \frac{1}{p_i}} \left(\frac{\ln \frac{8}{7}}{2(d-1)} \right)^{d-1} \right\} \\
&\geq \left(\frac{1}{15(d-1)} \right)^{d-1} \sqrt{\prod_{i=1}^{d-1} \frac{1}{p_i}}
\end{aligned}$$

where the inequality in the second line follows from $\ln(y) \leq 2(d-1) y^{\frac{1}{2(d-1)}}$ and the last line above is due to the fact that $|\Pi| \geq 1$ and $d \geq 2$. Now,

$$|\Pi| = \prod_{i=1}^{d-1} \frac{1}{p_i} \geq \left(\frac{16}{3} [15(d-1)]^{d-1} \right)^2$$

is sufficient for $\frac{|\Pi|}{C} \geq \frac{16}{3}$ which yields

$$\mathbb{E}_0[\tau] \geq \frac{1}{16\varepsilon^2} \frac{|\Pi|}{C} \log(1/2\delta) - \left(2T_{\max} + \frac{7}{12\varepsilon^2} \log(1/2\delta) \right) \frac{|\Pi|}{C} \Delta(T_{\max}) \,.$$

Since $\Delta(T_{\max})$ is the only term that depends on N, we can pick N large enough so that the first term dominates and

$$\mathbb{E}_0[\tau] \geq \frac{1}{17\varepsilon^2} \frac{|\Pi|}{C} \log(1/2\delta) \geq \frac{1}{17\varepsilon^2} \left(\frac{1}{15(d-1)} \right)^{d-1} \sqrt{\prod_{i=1}^{d-1} \frac{1}{p_i}} \log(1/2\delta).$$

It only remains to resolve the $1/\varepsilon^2$ to $1/\widetilde{\varepsilon}^2$. To that end, we now bound the probability of reaching the goal state by the end of the episode by

$$\Pr(G \leq H - 1) \geq \Pr\left(G \leq \sum_{i=1}^{d-1} \frac{2}{p_i} \ln \frac{2d}{1/2} \right) \geq \frac{1}{2} \,,$$

because by Lemma 21, the probability that the agent spends more than $\frac{2}{p_i} \ln \frac{2d}{1/2}$ time steps in states (i, b) or (i, g) is bounded by $\frac{1}{2d}$. Thus,

$$\varepsilon = \frac{4\widetilde{\varepsilon}}{\Pr(G \leq H - 1)} \leq 8\widetilde{\varepsilon}.$$

which yields the final bound

$$\mathbb{E}_0[\tau] \geq \frac{1}{1100\widetilde{\varepsilon}^2} \left(\frac{1}{15(d-1)} \right)^{d-1} \sqrt{\prod_{i=1}^{d-1} \frac{1}{p_i}} \log(1/2\delta).$$

Since bound on the stopping time holds on average over instances $M_{0,\phi}$, there must be at least one MDP instance for which the expected stopping time adheres to this lower-bound. This proves the desired adaptive lower bound. □

Lemma 21. *Let the progression probabilities $p_i = p$ for all $i \in [d-1]$ where $p \in (0,1)$. Further, let G denote the time step within the episode at which a goal step is reached. For any $\delta \in (0,1)$*

$$\Pr\Big(G \leq \frac{2d}{p} \ln \frac{1}{\delta}\Big) \geq 1 - \delta \,.$$

Proof. The event that G is at least $n+1$ is equivalent to at most $d-2$ state progressions within n trials which each happen with probability p. Let $X_i \in \{0,1\}$ be the indicator for a state progression at time i. Then

$$\begin{aligned}\Pr(G \geq n+1) &= \Pr\Big(\sum_{i=1}^{n} X_i \leq d-2\Big) = \Pr\Big(\sum_{i=1}^{n} X_i \leq np\big(1 - \big(\frac{d-2}{np} - 1\big)\big)\Big) \\ &\leq \exp\Big(-\frac{np}{2}\Big(\frac{d-2}{np} - 1\Big)^2\Big)\end{aligned}$$

by a multiplicative Chernoff bound. This yields for all $\delta \in (0,1)$

$$\Pr\Big(G \geq \frac{2d}{p} \ln 1/\delta\Big) \geq 1 - \delta \,.$$

□

E.4 Change of observation distributions

Lemma 22. *Let $\mathcal{F}_{i,h-1} = \sigma\big(\mathcal{F}_{i-1}, \{X_{i,h'}, A_{i,h'}, R_{i,h'}\}_{h' \in [h-1]}\big)$ be the sigma-field of everything observable up to before the h'th observation in episode i. Then*

$$\begin{aligned}\|\Pr_{0,\pi^*}(X_{i,h} | \mathcal{F}_{i,h-1}) - \Pr_{\textit{unif}}(X_{i,h})\|_1 &\leq 2\frac{|\mathcal{S}|Hi}{N} \\ \|\Pr_{\pi^*}(X_{i,h} | \mathcal{F}_{i,h-1}) - \Pr_{\textit{unif}}(X_{i,h})\|_1 &\leq 2\frac{|\mathcal{S}|Hi}{N} \\ \|\Pr_{0}(X_{i,h} | \mathcal{F}_{i,h-1}) - \Pr_{\textit{unif}}(X_{i,h})\|_1 &\leq 2\frac{|\mathcal{S}|Hi}{N} \,,\end{aligned}$$

where $\Pr_{\textit{unif}}(X_{i,h})$ is the uniform distribution over all possible observations $\mathcal{X}$.

Proof. We prove the statement for $\Pr_{0,\pi^*}$ but the others can be proven analogously.

Let $\mathcal{F}_{i,h-1} = \sigma\big(\mathcal{F}i-1, \{X_{i,h'}, A_{i,h'}, R_{i,h'}\}_{h' \in [h-1]}\big)$ be the sigma-field of everything observable up to before the h'th observation in episode i. Further, $\mathcal{F}'_{i,h-1} = \sigma\big(\mathcal{F}_{i,h-1}, \{S_{k,l}\}_{k \in [i-1], l \in [H]}, \{S_{i,l}\}_{l \in [h]}\big)$ is the sigma-field that in addition includes all latent state labels up to $S_{i,h}$.

Since $\mathcal{F}'_{i,h}$ determines the latent state mapping ϕ for the observations encountered so far but all assignment of the remaining observations remains equally likely, we can write the conditional distribution of observation $X_{i,h}$ in closed form as

$$\Pr_{0,\pi^*}\big(X_{i,h} = x | \mathcal{F}'_{i,h-1}\big) = \begin{cases} \frac{1}{N/|\mathcal{S}|} & \text{if } x \in \mathcal{X}^s_{\text{obs}} \\ 0 & \text{if } x \in \mathcal{X}_{\text{obs}} \setminus \mathcal{X}^s_{\text{obs}} \\ \big(1 - \frac{|\mathcal{X}^s_{\text{obs}}|}{N/|\mathcal{S}|}\big)\frac{1}{N - |\mathcal{X}_{\text{obs}}|} & \text{if } x \in \mathcal{X} \setminus \mathcal{X}_{\text{obs}} \end{cases}$$

where $\mathcal{X}_{\text{obs}}$ are all observations encountered so far and $\mathcal{X}^s_{\text{obs}}$ are all observations encountered in $S_{i,h}$ so far. Now

$$\big|\Pr_{0,\pi^*}\big(X_{i,h} = x | \mathcal{F}'_{i,h-1}\big) - \Pr_{\text{unif}}(X_{i,h} = x)\big| = \begin{cases} \frac{1}{N/|\mathcal{S}|} - \frac{1}{N} & \text{if } x \in \mathcal{X}^s_{\text{obs}} \\ \frac{1}{N} & \text{if } x \in \mathcal{X}_{\text{obs}} \setminus \mathcal{X}^s_{\text{obs}} \\ \Big|\frac{1}{N} - \big(1 - \frac{|\mathcal{X}^s_{\text{obs}}|}{N/|\mathcal{S}|}\big)\frac{1}{N - |\mathcal{X}_{\text{obs}}|}\Big| & \text{if } x \in \mathcal{X} \setminus \mathcal{X}_{\text{obs}} \end{cases}$$

and thus

$$\big\|\Pr_{0,\pi^*}\big(X_{i,h} | \mathcal{F}'_{i,h-1}\big) - \Pr_{\text{unif}}(X_{i,h})\big\|_1 \leq \frac{2\max\{|\mathcal{S}||\mathcal{X}^s_{\text{obs}}|, |\mathcal{X}_{\text{obs}}|\}}{N} \leq \frac{2|\mathcal{S}|T_{\max}H}{N} \,.$$

Since $\Pr_{0,\pi^*}(X_{i,h}|\mathcal{F}_{i,h-1}) = \mathbb{E}_{0,\pi^*}(\Pr_{0,\pi^*}(X_{i,h}|\mathcal{F}'_{i,h-1})|\mathcal{F}_{i,h-1})$ by marginalization, we also have

$$\|\Pr_{0,\pi^*}(X_{i,h}|\mathcal{F}_{i,h-1}) - \Pr_{\text{unif}}(X_{i,h})\|_1 \leq \frac{2|\mathcal{S}|T_{\max}H}{N}$$

which means that as long as $T_{\max} \ll N$, the conditional distribution of the current observations remains close to Uniform($\mathcal{X}$). □

Lemma 23. *Let $\mathcal{F}_i = \sigma\big(\{X_{k,h'}, A_{k,h'}, R_{k,h'}\}_{h'\in[H],k\in[i]}\big)$ denote the natural filtration at the end of episode i. Then*

$$\|\Pr_{\pi^*}(X_{i,1:H}|\mathcal{F}_{i-1}) - \Pr_{\textit{unif}}(X_{i,1:H})\|_1 \leq \frac{2|\mathcal{S}|H^2 i}{N}$$
$$\|\Pr_{0}(X_{i,1:H}|\mathcal{F}_{i-1}) - \Pr_{\textit{unif}}(X_{i,1:H})\|_1 \leq \frac{2|\mathcal{S}|H^2 i}{N}\,,$$

where $\Pr_{\textit{unif}}(X_{i,1:H})$ is the product distribution of uniform distributions over all possible observations $\mathcal{X}$.

Proof. The random variables $X_{i,1:H}$ are $\mathcal{F}_i$-measurable. We can therefore consider any event $A \in \mathcal{F}_i$ and show that

$$|\Pr_{\pi^*}(A|\mathcal{F}_{i-1}) - \Pr_{\text{unif}}(A)| \leq \frac{|\mathcal{S}|H^2 i}{N}$$
$$|\Pr_{0}(A|\mathcal{F}_{i-1}) - \Pr_{\text{unif}}(A)| \leq \frac{|\mathcal{S}|H^2 i}{N}$$

analogously to Lemma 24 below. The result then follows immediately from the identity of ℓ_1 norm and total variation. □

Lemma 24. *Let $A \in \mathcal{F}_{T_{\max}}$ be any event that is $\mathcal{F}_{T_{\max}}$-measurable, where $\mathcal{F}_{T_{\max}}$ is the sigma-field induced by everything up to $T_{\max}$ episodes. Then*

$$|\Pr_0(A) - \Pr_{0,\pi^*}(A)| \leq \Delta(T_{\max}) = \frac{4T_{\max}^2 H^2|\mathcal{S}|}{N}\,.$$

Proof. Denote by $\Pr_{0,\pi^*}^{t,h,\text{unif}}$ the distribution that matches $\Pr_{0,\pi^*}$ but where all observations after the h'th observation in episode t are drawn uniformly random from $\mathcal{X}$. First, since $A \in \mathcal{F}_{T_{\max}}$ and $\Pr_{0,\pi^*}(B) = \Pr_{0,\pi^*}^{T_{\max},\text{unif}}(B)$ for all events $B \in \mathcal{F}_{T_{\max}}$, we have

$$\Pr_{0,\pi^*}(A) = \Pr_{0,\pi^*}^{T_{\max},H,\text{unif}}(A)\,.$$

We now peel off one time step at a time by showing that $\left|\Pr_{0,\pi^*}^{t,h,\text{unif}}(A) - \Pr_{0,\pi^*}^{t,h+1,\text{unif}}(A)\right| \leq \frac{2|\mathcal{S}|Ht}{N}$. By the definition of these probabilities, the following chain of equations holds:

$$\begin{aligned}
&\Pr_{0,\pi^*}^{t,h,\text{unif}}(A) \\
&= \mathbb{E}_{0,\pi^*}^{t,h,\text{unif}}\Big[\Pr_{0,\pi^*}^{t,h,\text{unif}}(A|X_{t,h},\mathcal{F}_{t,h-1})\Big] = \mathbb{E}_{0,\pi^*}^{t,h,\text{unif}}\Big[\Pr_{0,\pi^*}^{t,h-1,\text{unif}}(A|X_{t,h},\mathcal{F}_{t,h-1})\Big] \\
&= \mathbb{E}_{0,\pi^*}^{t,h,\text{unif}}\Big[\sum_{x\in\mathcal{X}} \Pr_{0,\pi^*}^{t,h-1,\text{unif}}(A|X_{t,h}=x,\mathcal{F}_{t,h-1})\Pr_{0,\pi^*}^{t,h,\text{unif}}(X_{t,h}=x|\mathcal{F}_{t,h-1})\Big] \\
&= \mathbb{E}_{0,\pi^*}^{t,h,\text{unif}}\Big[\sum_{x\in\mathcal{X}} \Pr_{0,\pi^*}^{t,h-1,\text{unif}}(A|X_{t,h}=x,\mathcal{F}_{t,h-1})\Pr_{0,\pi^*}^{t,h-1,\text{unif}}(X_{t,h}=x|\mathcal{F}_{t,h-1})\Big] \\
&\quad + \mathbb{E}_{0,\pi^*}^{t,h,\text{unif}}\Big[\sum_{x\in\mathcal{X}} \Pr_{0,\pi^*}^{t,h-1,\text{unif}}(A|X_{t,h}=x,\mathcal{F}_{t,h-1})\Big(\Pr_{0,\pi^*}^{t,h,\text{unif}}(X_{t,h}=x|\mathcal{F}_{t,h-1}) - \Pr_{0,\pi^*}^{t,h-1,\text{unif}}(X_{t,h}=x|\mathcal{F}_{t,h-1})\Big)\Big] \\
&= \Pr_{0,\pi^*}^{t,h-1,\text{unif}}(A) + \mathbb{E}_{0,\pi^*}^{t,h,\text{unif}}\Big[\sum_{x\in\mathcal{X}} \Pr_{0,\pi^*}^{t,h-1,\text{unif}}(A|X_{t,h}=x,\mathcal{F}_{t,h-1})(\Pr_{0,\pi^*}(X_{t,h}=x|\mathcal{F}_{t,h-1}) - \Pr_{\text{unif}}(X_{t,h}=x))\Big]\,.
\end{aligned}$$

Thus, by rearranging terms, we have

$$|\Pr_{0,\pi^*}^{t,h,\text{unif}}(A) - \Pr_{0,\pi^*}^{t,h-1,\text{unif}}(A)| \leq \mathbb{E}_{0,\pi^*}^{t,h,\text{unif}}\big[\|\Pr_{0,\pi^*}(X_{t,h}|\mathcal{F}_{t,h-1}) - \Pr_{\text{unif}}(X_{t,h})\|_1\big] \leq \frac{2|\mathcal{S}|Ht}{N}\,,$$

where the last inequality follows from Lemma 22. We now consider

$$\begin{aligned}\Pr_{0,\pi^*}(A) - \Pr_{0,\pi^*}^{1,0,\text{unif}}(A) &= \Pr_{0,\pi^*}^{T_{\max},H,\text{unif}}(A) - \Pr_{0,\pi^*}^{1,0,\text{unif}}(A) \\ &= \sum_{h=1}^{H}\sum_{t=1}^{T_{\max}} \Pr_{0,\pi^*}^{t,h,\text{unif}}(A) - \Pr_{0,\pi^*}^{t,h-1,\text{unif}}(A)\end{aligned}$$

where $\Pr_{0,\pi^*}^{t,0,\text{unif}} = \Pr_{0,\pi^*}^{t-1,H,\text{unif}}$ and apply the bound to each term to arrive at

$$|\Pr_{0,\pi^*}(A) - \Pr_{0,\pi^*}^{1,0,\text{unif}}(A)| \leq \frac{2T_{\max}^2 H^2 |\mathcal{S}|}{N} .$$

Note that for the distribution $\Pr_{0,\pi^*}^{1,0,\text{unif}}$ all observations are drawn uniformly at random and the rewards do not depend on the actions. Thus, $\Pr_{0,\pi^*}^{1,0,\text{unif}} = \Pr_{0}^{1,0,\text{unif}}$ and we can derive analogously to above that

$$|\Pr_{0}(A) - \Pr_{0}^{1,0,\text{unif}}(A)| = |\Pr_{0}(A) - \Pr_{0,\pi^*}^{1,0,\text{unif}}(A)| \leq \frac{2T_{\max}^2 H^2 |\mathcal{S}|}{N} .$$

Combining both bounds using the triangle inequality yields the desired statement

$$|\Pr_{0}(A) - \Pr_{0,\pi^*}(A)| \leq \frac{4T_{\max}^2 H^2 |\mathcal{S}|}{N} .$$

□

E.5 Bounds on expected policy matches per episode

Lemma 25 (Bound on expected policy matches with equal p_i)**.** *Let $\pi\colon \mathcal{X} \mapsto \mathcal{A}$ any policy (that does not need to be in the given policy class Π) and $H \geq 62d$. Further, set $|\Pi| = (H/d)^d$, and $p = \frac{d}{H}$. Then*

$$\mathbb{E}_{\textit{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\}\Big] \leq \Big(41\log(H/d)\Big)^d H + 2 ,$$

where $\Pr_{\textit{unif}}$ draws all H observations $X_{1:H}$ i.i.d. from $\text{Uniform}(\mathcal{X})$ and G as usual.

Proof. For any $h \in \mathbb{N}$ with $d \leq h \leq H$, the following holds:

$$\begin{aligned}&\mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\}\Big] \\ &= \Pr(G \leq h)\mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\} \mid G \leq h\Big] \\ &\quad + \Pr(G > h)\mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\} \mid G > h\Big] \\ &\leq |\Pi|\Pr(G \leq h) + \mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:h} = \pi(X_{1:h})\}\Big] \\ &\leq (h-d+1)|\Pi|\Big(\frac{2peh}{d}\Big)^{d-1} + \mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:h} = \pi(X_{1:h})\}\Big] ,\end{aligned}$$

where the last inequality applies Lemma 27. Note that by the construction of Π, there can only be one policy in Π which agrees with π on more than $\frac{7}{8}N$ observations in $\mathcal{X}$. With all other policies, π has to disagree on at least $1/8$ fraction of all possible observations. To see this, assume that there were two policies $\pi_1 \neq \pi_2$ in Π for which $\|\pi - \pi_i\| < N/8$. Then by triangle inequality $\|\pi_1 - \pi_2\| < N/4$ which contradicts the construction of Π. Thus, we can further bound the quantity of interest as

$$\mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\}\Big] \leq (h-d+1)|\Pi|\Big(\frac{2peh}{d}\Big)^{d-1} + 1 + (|\Pi|-1)\Big(\frac{7}{8}\Big)^h$$

$$\leq |\Pi| h \Big(\frac{2peh}{d}\Big)^{d-1} + 1 + |\Pi| \exp(-h \log(8/7)) \,. \tag{65}$$

We use the worst-case choices of $|\Pi|$, p and h as

$$|\Pi| = \left(\frac{H}{d}\right)^d, \qquad h = \frac{\log|\Pi|}{\log(8/7)}, \qquad p = \frac{d}{H} \,.$$

Since the RHS of Equation 65 is non-decreasing in p and Π, a bound with these exact values is also valid when p and Π are smaller. Note that under these choices $H \geq 62d$ which implies $\frac{H}{d} \geq 15 \ln \frac{H}{d}$ is sufficient for $h \leq H - 1$.

With these choices, the last term of Equation 65 is bounded by 1, i.e., $|\Pi| \exp\big(-h \log \frac{8}{7}\big) = 1$, and the first term is bounded as

$$\begin{aligned} |\Pi| h \big(\frac{2peh}{d}\big)^{d-1} &= \big(\frac{2e}{\log(8/7)}\big)^d \big(\frac{p}{d}\big)^{d-1} |\Pi| (\log|\Pi|)^d \\ &\leq 41^d \big(\frac{Hd}{d} \log \frac{H}{d}\big)^d \big(\frac{p}{d}\big)^{d-1} \leq \big(41 \log(H/d)\big)^d H \,. \end{aligned}$$

□

Lemma 26 (Bound on expected policy matches with arbitrary p_i)**.** *Let the probability for progressions be $p_1, \ldots, p_{d-1}$ and let $\pi \colon \mathcal{X} \mapsto \mathcal{A}$ any policy (that does not need to be in the given policy class Π). Further, assume that*

$$|\Pi| \leq \exp\Big(\frac{H}{2} \log \frac{8}{7}\Big).$$

Then

$$\mathbb{E}_{\mathit{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\}\Big] \leq 2 + |\Pi| \prod_{i=1}^{d-1} \Big(p_i \frac{\log|\Pi|}{\log(8/7)}\Big) \,,$$

where $\Pr_{\mathit{unif}}$ draws all H observations $X_{1:H}$ i.i.d. from $\mathrm{Uniform}(\mathcal{X})$ and G as usual.

Proof. For any $h \in \mathbb{N}$ with $d \leq h \leq H$, the following holds:

$$\begin{aligned} \mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\}\Big] &= \Pr(G \leq h) \mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\} \;\mid G \leq h\Big] \\ &\quad + \Pr(G > h) \mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\} \;\mid G > h\Big] \\ &\leq |\Pi| \Pr(G \leq h) + \mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:h} = \pi(X_{1:h})\}\Big] \\ &\leq |\Pi| h^{d-1} \prod_{i=1}^{d-1} p_i + \mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:h} = \pi(X_{1:h})\}\Big] \,, \end{aligned}$$

where the last inequality applies Lemma 28. Note that by the construction of Π, there can only be one policy in Π which agrees with π on more than $\frac{7}{8}N$ observations in $\mathcal{X}$. With all other policies, π has to disagree on at least $1/8$ fraction of all possible observations. To see this, assume that there were two policies $\pi_1 \neq \pi_2$ in Π for which $\|\pi - \pi_i\| < N/8$. Then by triangle inequality $\|\pi_1 - \pi_2\| < N/4$ which contradicts the construction of Π. Thus, we can further bound the quantity of interest as

$$\begin{aligned} \mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\}\Big] &\leq |\Pi| h^{d-1} \prod_{i=1}^{d-1} p_i + 1 + (|\Pi| - 1)\left(\frac{7}{8}\right)^h \\ &\leq |\Pi| h^{d-1} \prod_{i=1}^{d-1} p_i + 1 + |\Pi| \exp(-h \log(8/7)) \,. \end{aligned} \tag{66}$$

We now set $h = \frac{\log|\Pi|}{\log(8/7)}$ which gives

$$\mathbb{E}_{\text{unif}}\Big[\sum_{\pi^*} \mathbb{1}\{\pi^*(X_{1:G} = \pi(X_{1:G})\}\Big] \leq 2 + |\Pi| \prod_{i=1}^{d-1} \Big(p_i \frac{\log|\Pi|}{\log(8/7)}\Big) \,. \tag{67}$$

□

Lemma 27. *Let the progression probabilities $p_i = p$ for all $i \in [d-1]$ where $p \in (0,1)$. The time step G within the episode at which a goal step is reached satisfies*

$$\Pr(G \leq h) \leq (h - d + 1)\Big(\frac{2peh}{d}\Big)^{d-1}.$$

Proof. For $G = i$, there must be exactly $d-1$ progressions in the $i-1$ previous time steps, each happening with probability p. Therefore

$$\Pr(G = i) = \binom{i-1}{d-1} p^{d-1}(1-p)^{i-d}.$$

Thus,

$$\begin{aligned}
\Pr(G \leq h) &= \sum_{i=d}^{h} \Pr(G = i) = \sum_{i=d}^{h} \binom{i-1}{d-1} p^{d-1}(1-p)^{h-d} \\
&\leq \sum_{i=d}^{h} \binom{i-1}{d-1} p^{d-1} \leq \sum_{i=d}^{h} \Big(\frac{e(i-1)}{d-1}\Big)^{d-1} p^{d-1}, \\
&\leq (h - d + 1)\Big(\frac{2peh}{d}\Big)^{d-1},
\end{aligned}$$

where the first inequality in the above is given by ignoring terms smaller than one, and the second inequality is due to the fact that any n, k, we have $\binom{n}{k} \leq (en/k)^k$ for $0 \leq k \leq n$. □

Lemma 28. *Let the probability for progressions be $p_1, \ldots, p_{d-1}$. The time step G within the episode at which a goal step is reached then satisfies*

$$\Pr(G \leq h) \leq h^{d-1} \prod_{i=1}^{d-1} p_i$$

Proof. For the event $G \leq h$ to happen, there must have been a progression in each of the $d-1$ states within h trials. Therefore

$$\Pr(G \leq h) \leq \prod_{i=1}^{d-1} (1 - (1-p_i)^{h-1}) \leq \prod_{i=1}^{d-1} ((h-1)p_i) \leq \prod_{i=1}^{d-1} (hp_i) \,.$$

□