PROVABLE RL WITH EXOGENOUS DISTRACTORS VIA MULTISTEP INVERSE DYNAMICS

Yonathan Efroni[1], **Dipendra Misra**[1], **Akshay Krishnamurthy**[1], **Alekh Agarwal**[2†], **John Langford**[1]
[1]Microsoft Research, New York, NY
[2]Google

ABSTRACT

Many real-world applications of reinforcement learning (RL) require the agent to deal with high-dimensional observations such as those generated from a megapixel camera. Prior work has addressed such problems with representation learning, through which the agent can provably extract endogenous, latent state information from raw observations and subsequently plan efficiently. However, such approaches can fail in the presence of temporally correlated noise in the observations, a phenomenon that is common in practice. We initiate the formal study of latent state discovery in the presence of such *exogenous* noise sources by proposing a new model, the Exogenous Block MDP (EX-BMDP), for rich observation RL. We start by establishing several negative results, by highlighting failure cases of prior representation learning based approaches. Then, we introduce the Predictive Path Elimination (PPE) algorithm, that learns a generalization of inverse dynamics and is provably sample and computationally efficient in EX-BMDPs when the endogenous state dynamics are near deterministic. The sample complexity of PPE depends polynomially on the size of the latent endogenous state space while not directly depending on the size of the observation space, nor the exogenous state space. We provide experiments on challenging exploration problems which show that our approach works empirically.

1 INTRODUCTION

In many real-world applications such as robotics there can be large disparities in the size of agent's observation space (for example, the image generated by agent's camera), and a much smaller latent state space (for example, the agent's location and orientation) governing the rewards and dynamics. This size disparity offers an opportunity: how can we construct reinforcement learning (RL) algorithms which can learn an optimal policy using samples that scale with the size of the latent state space rather than the size of the observation space? Several families of approaches have been proposed based on solving various ancillary prediction problems including autoencoding (Tang et al., 2017; Hafner et al., 2019), inverse modeling (Pathak et al., 2017; Burda et al., 2018), and contrastive learning (Laskin et al., 2020) based approaches. These works have generated some significant empirical successes, but are there provable (and hence more reliable) foundations for their success? More generally, what are the right principles for learning with latent state spaces?

In real-world applications, a key issue is robustness to noise in the observation space. When noise comes from the observation process itself, such as due to measurement error, several approaches have been recently developed to either explicitly identify (Du et al., 2019; Misra et al., 2020; Agarwal et al., 2020a) or implicitly leverage (Jiang et al., 2017) the presence of latent state structure for provably sample-efficient RL. However, in many real-world scenarios, the observations consist of many elements (e.g. weather, lighting conditions, etc.) with temporally correlated dynamics (see e.g. Figure 1 and the example below) that are entirely independent of the agent's actions and rewards. The temporal dynamics of these elements precludes us from treating them as uncorrelated noise, and as such, most previous approaches resort to modeling their dynamics. However, this is clearly wasteful as these elements have no bearing on the RL problem being solved.

[†]Work was done while the author was at Microsoft Research.
{yefroni, dimisra, akshaykr, jcl}@microsoft.com, alekhagarwal@google.com

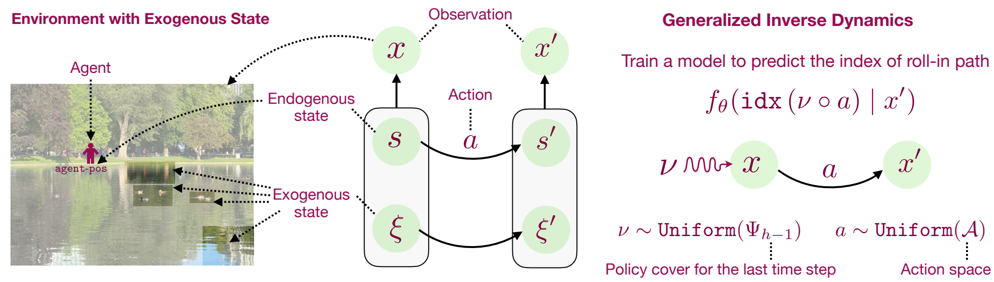


Figure 1: **Left:** An agent is walking next to a pond in a park and observes the world as an image. The world consists of a latent endogenous state, containing variable such as agent's position, and a much larger latent exogenous state containing variables such as motion of ducks, ripples in the water, etc. **Center:** Graphical model of the EX-BMDP. **Right:** PPE learns a generalized form of inverse dynamics that recovers the endogenous state.

As an example, consider the setting in Figure 1. An agent is walking in a park on a lonely sidewalk next to a pond. The agent's observation space is the image generated by its camera, the latent endogenous state is its position on the sidewalk, and the exogenous noise is provided by motion of ducks, swaying of trees and changes in lighting conditions, typically unaffected by the agent's actions. While there is a line of recent empirical work that aims to remove causally irrelevant aspects of the observation (Gelada et al., 2019; Zhang et al., 2020), theoretical treatment is quite limited (Dietterich et al., 2018) and no prior works address sample-efficient learning with provable guarantees. Given this, the key question here is:

How can we learn using an amount of data scaling with just the size of the endogenous *latent state, while ignoring the temporally correlated* exogenous *observation noise?*

We initiate a formal treatment of RL settings where the learner's observations are jointly generated by a latent endogenous state and an uncontrolled exogenous state, which is unaffected by the agent's actions and does not affect the agent's task. We study a subset of such problems called Exogenous Block MDPs (EX-BMDPs), where the endogenous state is discrete and decodable from the observations. We first highlight the challenges in solving EX-BMDPs by illustrating the failures of many prior representation learning approaches (Pathak et al., 2017; Misra et al., 2020; Jiang et al., 2017; Agarwal et al., 2020a; Zhang et al., 2020). These failure happen either due to creating too many latent states, such as one for each combination of ducks and passers-by in the example above leading to sample inefficiency in exploration, or due to lack of exhaustive exploration.

We identify one recent approach developed by Du et al. (2019) with favorable properties for EX-BMDPs with near-deterministic latent state dynamics. In Section 4 and Section 5, we develop a variation of their algorithm and analyze its performance. The algorithm, called Path Prediction and Elimination (PPE), learns a form of *multi-step inverse dynamics* by predicting the identity of the path that generates an observation. For near-deterministic EX-BMDPs, we prove that PPE successfully explores the environment using $O((SA)^2 H \log(|\mathcal{F}|/\delta))$ samples where S is the size of the latent *endogenous* state space, A is the number of actions, H is the horizon and $\mathcal{F}$ is a function class employed to solve a maximum likelihood problem. Several prior works (Gregor et al., 2016; Paster et al., 2020) have also considered a multi-step inverse dynamics approach to learn a near optimal policy. Yet, these works do not consider the EX-BMDP model. Further, it is unknown whether these algorithms have provable guarantees, as PPE. Theoretical analysis of the performance of these algorithms in the presence of exogenous noise is an interesting future work direction.

Empirically, in Section 6, we demonstrate the performance of PPE and various prior baselines in a challenging exploration problem with exogenous noise. We show that baselines fail to decode the endogenous state as well as learning a good policy. We further, show that PPE is able to recover the latent endogenous model in a visually complex navigation problem, in accordance with the theory.

2 Exogenous Block MDP Setting

We introduce a novel *Exogenous Block Markov Decision Process* (EX-BMDP) setting to model systems with exogenous noise. We describe notations before formalizing the EX-BMDP model.

Notations. For a given set $\mathcal{U}$, we use $\Delta(\mathcal{U})$ to denote the set of all probability distributions over $\mathcal{U}$. For a given natural number $N \in \mathbb{N}$, we use the notation $[N]$ to denote the set $\{1, 2, \cdots, N\}$. Lastly, for a probability distribution $p \in \Delta(\mathcal{U})$, we define its support as $supp(p) = \{u \mid p(u) > 0, u \in \mathcal{U}\}$.

We start with describing the Block Markov Decision Process (BMDP) Du et al. (2019). This process consists of a finite set of observations $\mathcal{X}$, a set of *latent* states $\mathcal{Z}$ with cardinality Z, a finite set of actions $\mathcal{A}$ with cardinality A, a transition function $T : \mathcal{Z} \times \mathcal{A} \to \Delta(\mathcal{Z})$, an emission function $q : \mathcal{Z} \times \mathcal{A} \to \Delta(\mathcal{X})$, a reward function $R : \mathcal{X} \times \mathcal{A} \to [0, 1]$, a horizon $H \in \mathbb{N}$, and a start state distribution $\mu \in \Delta(\mathcal{Z})$. The agent interacts with the environment by repeatedly generating H-step trajectories $(z_1, x_1, a_1, r_1, \cdots, z_H, x_H, a_H, r_H)$ where $z_1 \sim \mu(\cdot)$ and for every $h \in [H]$ we have $x_h \sim q(\cdot \mid z_h)$, $r_h = R(x_h, a_h)$, and if $h < H$, then $z_{h+1} \sim T(\cdot \mid z_h, a_h)$. The agent does not observe the states $(z_1, \cdots, z_H)$, instead receiving only the observations $(x_1, \cdots, x_H)$ and rewards $(r_1, \cdots, r_H)$. We assume that the emission distributions of any two latent states are disjoint, usually referred as *the block assumption*: $supp(q(\cdot \mid z_1)) \cap supp(q(\cdot|z_2)) = \emptyset$ when $z_1 \neq z_2$. The agent chooses actions using a policy $\pi : \mathcal{X} \to \Delta(\mathcal{A})$. We also define the set of non-stationary policies $\Pi_{\text{NS}} = \Pi^H$ as a H-length tuple, with $(\pi_1, \cdots, \pi_H) \in \Pi_{\text{NS}}$ denoting that the action at time step h is taken as $a_h \sim \pi_h(\cdot \mid x_h)$. The value $V(\pi)$ of a policy π is the expected episodic sum of rewards $V(\pi) := \mathbb{E}_\pi[\sum_{h=1}^{H} R(x_h, a_h)]$. The optimal policy is given by $\pi^\star = \arg\max_{\pi \in \Pi_{\text{NS}}} V(\pi)$. We denote by $\mathbb{P}_h(x|\pi)$ the probability distribution over observations x at time step h when following a policy π. Lastly, we refer to an *open loop* policy as an element in all $\mathcal{A}^H$ sequences of actions. An open loop policy follows a pre-determined sequence of actions $\{a_1, .., a_H\}$ for H time steps, unaffected by state information.

Given the aforementioned definitions, we define an EX-BMDP as follows:

Definition 1 (Exogenous Block Markov Decision Processes). *An EX-BMDP is a BMDP such that the latent state can be decoupled into two parts $z = (s, \xi)$ where $s \in \mathcal{S}$ is the endogenous state and $\xi \in \Xi$ is the exogenous state. For $z \in \mathcal{Z}$ the initial distribution and transition functions are decoupled, that is: $\mu(z) = \mu(s)\mu_\xi(\xi)$, and $T(z' \mid z, a) = T(s' \mid s, a)T_\xi(\xi' \mid \xi)$.*

The observation space $\mathcal{X}$ can be arbitrarily large to model which could be a high-dimensional real vector denoting an image, sound, or haptic data in an EX-BMDP. The endogenous state s captures the information that can be manipulated by the agent. Figure 1, center, visualizes the transition dynamics factorization. We assume that the set of all endogenous states $\mathcal{S}$ is finite with cardinality S. The exogenous state ξ captures all the other information that the agent cannot control and does not affect the information it can manipulate. Again, we make no assumptions on the exogenous dynamics nor on its cardinality $|\Xi|$ which may be arbitrarily large. We note that the block assumption of the EX-BMDP implies the existence of two inverse mappings: $\phi^\star : \mathcal{X} \to \mathcal{S}$ to map an observation to its endogenous state, and $\phi^\star_\xi : \mathcal{X} \to \Xi$ to map it to its exogenous state.

Justification of assumptions. The block assumption has been made by prior work (e.g., Du et al. (2019), Zhang et al. (2020)) to model many real-world settings where the observation is *rich*, i.e., it contains enough information to decode the latent state. The decoupled dynamics assumption made in the EX-BMDP setting is a natural way to characterize exogenous noise; the type of noise that is not affected by our actions nor affects the endogenous state but may have non-trivial dynamic. This decoupling captures the movement of ducks, captured in the visual field of the agent in Figure 1, and many additional exogenous processes (e.g., movement of clouds in a navigation task).

Goal. Our formal objective is reward-free learning. We wish to find a set of policies, we call a *policy cover*, that can be used to explore the entire state space. Given a policy cover, and for any reward function, we can find a near optimal policy by applying dynamic programming (e.g., Bagnell et al. (2004)), policy optimization (e.g., Kakade and Langford (2002); Agarwal et al. (2020b); Shani et al. (2020)) or value (e.g., Antos et al. (2008)) based methods.

Definition 2 (α-policy cover). *Let Ψ_h be a finite set of non-stationary policies. We say Ψ_h is an α-policy cover for the h^{th} time step if for all $z \in \mathcal{Z}$ it holds that $\max_{\pi \in \Psi_h} \mathbb{P}_h(z|\pi) \geq \max_{\pi \in \Pi_{NS}} \mathbb{P}_h(z|\pi) - \alpha$. If $\alpha = 0$ we call Ψ_h a policy cover.*

For standard BMDPs the policy cover is simply the set of policies that reaches each latent state of the BMDP (Du et al., 2019; Misra et al., 2020; Agarwal et al., 2020a). Thus, for a BMDP, the cardinality of the policy cover scales with $|\mathcal{Z}|$. The structure of EX-BMDPs allows to reduce the size of the policy cover significantly to $|\mathcal{S}| \ll |\mathcal{Z}| = |\mathcal{S}|\,|\Xi|$ when the size of the exogenous state space is large. Specifically, we show that the set of policies that reach each *endogenous* state, and *do not depend on the exogenous* part of the state is also a policy cover (see Appendix B, Proposition 4).

3 FAILURES OF PRIOR APPROACHES

We now describe the limitation of prior RL approaches in the presence of exogenous noise. We provide an intuitive analysis over here, and defer a formal statement and proof to Appendix A.

Limitation of Noise-Contrastive learning. Noise-contrastive learning has been used in RL to learn a state abstraction by exploiting temporal information. Specifically, the HOMER algorithm (Misra et al., 2020) trains a model to distinguish between *real* and *imposter* transitions. This is done by collecting a dataset of quads (x, a, x', y) where $y = 1$ means the transition was (x, a, x') was observed and $y = 0$ means that (x, a, x') was not observed. HOMER then trains a model $p_\theta(y \mid x, a, \phi_\theta(x'))$ with parameters θ, on the dataset, by predicting whether a given pair of transition was observed or not. This provides a state abstraction $\phi_\theta : \mathcal{X} \to \mathbb{N}$ for exploring the environment. HOMER can provably solve Block MDPs. Unfortunately, in the presence of exogenous noise, HOMER distinguishes between two transitions that represent transition between the same latent endogenous states but different exogenous states. In our walk in the park example, even if the agent moves between same points in two transitions, the model maybe able to tell these transitions apart by looking at the position of ducks which may have different behaviour in the two transitions. This results in the HOMER creating $\mathcal{O}(|\mathcal{Z}|)$ many abstract states. We call this the *under-abstraction* problem.

Limitation of Inverse Dynamics. Another common approach in empirical works is based on modeling the inverse dynamics of the system, such as the ICM module of Pathak et al. (2017). In such approaches, we learn a representation by using consecutive observations to predict the action that was taken between them. Such a representation can ignore all information that is not relevant for action prediction, which includes all exogenous/uncontrollable information. However, it can also ignore controllable information. This may result in a failure to sufficiently explore the environment. In this sense, inverse dynamics approaches result in an *over-abstraction* problem where observations from different endogenous states can be mapped to the same abstract state. The over-abstraction problem was described at Misra et al. (2020), when the starting state is random. In Appendix A.3 we show inverse dynamics may over-abstract when the initial starting state is deterministic.

Limitation of Bisimulation. Zhang et al. (2020) proposed learning a bisimulation metric to learn a representation which is invariant to exogenous noise. Unfortunately, it is known that bisimulation metric cannot be learned in a sample-efficient manner (Modi et al. (2020), Proposition B.1). Intuitively, when the reward is same everywhere, then bisimulation merges all states into a single abstract state. This creates an *over-abstraction* problem in sparse reward settings, since the agent can falsely merge all states into a single abstract state until it receives a non-trivial reward.

Bellman rank might depend on $|\Xi|$. The Bellman rank was introduced in Jiang et al. (2017) as a complexity measure for the learnability of an RL problem with function approximations. To date, most of the learnable RL problems have a small Bellman rank. However, we show in Appendix A that Bellman rank for EX-BMDP can scale as $\mathcal{O}(|\Xi|)$. This shows that EX-BMDP is a highly non-trivial setting as we don't even have sample-efficient algorithms regardless of computationally-efficient.

In Appendix A we also describe the failures of `FLAMBE` (Agarwal et al., 2020a)) and autoencoding based approaches (Tang et al., 2017).

4 REINFORCEMENT LEARNING FOR EX-BMDPS

In this section, we present an algorithm *Predictive Path Elimination* (`PPE`) that we later show can provably solve any EX-BMDP with nearly deterministic dynamics and start state distribution of the endogenous state, while making no assumptions on the dynamics or start state distribution of the exogenous state (Algorithm 1). Before describing `PPE`, we highlight that `PPE` can be thought of as

Algorithm 1 PPE(δ, η): Predictive Path Elimination

1: Set $\Psi_1 = \{\perp\}$, stochasticity level $\eta \leq \frac{1}{4SH}$ //$\perp$ denotes an empty path
2: **for** $h = 2, \ldots, H$ **do**
3: Set $N = 16\left(|\Psi_{h-1} \circ A|\right)^2 \log\left(\frac{|\mathcal{F}||\Psi_{h-1}|AH}{\delta}\right)$
4: Collect a dataset $\mathcal{D}$ of N *i.i.d.* tuples (x, υ) where $\upsilon \sim \texttt{Unf}(\Psi_{h-1} \circ \mathcal{A})$ and $x \sim \mathbb{P}(x_h \mid \upsilon)$.
5: Solve multi-class classification problem: $\hat{f}_h = \arg\max_{f \in \mathcal{F}} \sum_{(x,\upsilon) \in \mathcal{D}} \ln f(\text{idx}(\upsilon) \mid x)$.
6: **for** $1 \leq i < j \leq |\Psi_{h-1} \circ \mathcal{A}|$ **do**
7: Calculate the path prediction gap: $\widehat{\Delta}(i, j) = \frac{1}{N} \sum_{(x,\upsilon) \in \mathcal{D}} \left|\hat{f}_h(i|x) - \hat{f}_h(j|x)\right|$.
8: If $\widehat{\Delta}(i, j) \leq \frac{5/8}{|\Psi_{h-1} \circ \mathcal{A}|}$, then eliminate path υ with $\text{idx}(\upsilon) = j$. //υ_i and υ_j visit same state
9: Ψ_h is defined as the set of all paths in $\Psi_{h-1} \circ \mathcal{A}$ that have not been eliminated in line 8.

a computationally-efficient and simpler alternative to Algorithm 4 of Du et al. (2019) who studied rich-observation setting without exogenous noise.[1]

PPE performs iterations over the time steps $h \in \{2, \cdots, H\}$. In the h^{th} iteration, it learns a policy cover Ψ_h for time step h containing open-loop policies. This is done by first augmenting the policy cover for previous time step by one step. Formally, we define $\Upsilon_h = \Psi_{h-1} \circ \mathcal{A} = \{\pi \circ a \mid \pi \in \Psi_{h-1}, a \in \mathcal{A}\}$ where $\pi \circ a$ is an open-loop policy that follows π till time step $h - 1$ and then takes action a. Since we assume the transition dynamics to be near-deterministic, therefore, we know that there exists a policy cover for time step h that is a subset of Υ_h and whose size is equal to the number of reachable states at time step h. Further, as the transitions are near-deterministic, we refer to an open-loop policy as a path, as we can view the policy as tracing a path in the latent transition model. PPE works by eliminating paths in Υ_h so that we are left with just a single path for each reachable state. This is done by collecting a dataset $\mathcal{D}$ of tuples (x, υ) where υ is a uniformly sampled from Υ_h and $x \sim \mathbb{P}_h(x \mid \upsilon)$ (line 4). We train a classifier $\hat{f}_h$ using $\mathcal{D}$ by predicting the index $\text{idx}(\upsilon)$ of the path υ from the observation x (line 5). Index of paths in Υ_h are computed with respect to Υ_h and remain fixed throughout training. Intuitively, if $\hat{f}_h(i \mid x)$ is sufficiently large, then we can hope that the path υ_i visits the state $\phi^\star(x)$. Further, we can view this prediction problem as learning a multistep inverse dynamics model since the open-loop policy contains information about all previous actions and not just the last action. For every pair of paths in Υ_h, we first compute a path prediction gap $\widehat{\Delta}$(line 7). If the gap is too small, we show it implies that these paths reach the same endogenous state, hence we can eliminate a single redundant path from this pair (line 8). Finally, Ψ_h is defined as the set of all paths in Υ_h which were not eliminated. PPE reduces RL to performing H standard classification problems. Further, the algorithm is very simple and in practice requires just a single hyperparameter (N). We believe these properties will make it well-suited for many problems.

Recovering an endogenous state decoder. We can recover a endogenous state decoder $\hat{\phi}_h$ for each time step $h \in \{2, 3, \cdots, H\}$ directly from $\hat{f}_h$ as shown below:

$$\hat{\phi}_h(x) = \min\left\{i \mid \hat{f}_h(i \mid x) \geq \max_j \hat{f}_h(j \mid x) - \mathcal{O}(1/|\Upsilon_h|), i \in [|\Upsilon_h|]\right\}.$$

Intuitively, this assigns the observation to the path with smallest index that has the highest chance of visiting x, and therefore, $\phi^\star(x)$. We are implicitly using the decoder for exploring, since we rely on using $\hat{f}_h$ for making planning decisions. We will evaluate the accuracy of this decoder in Section 6.

Recovering the latent transition dynamics. PPE can also be used to recover a latent endogenous transition dynamics. The direct way is to use the learned decoder $\hat{\phi}_h$ along with episodes collected by PPE during the course of training and do count-based estimation. However, for most problems, recovering an approximate deterministic transition dynamics suffices, which can be directly read

[1] Alg. 4 has time complexity of $\mathcal{O}(S^4A^4H)$ compared to $\mathcal{O}(S^3A^3H)$ for PPE. Furthermore, Alg. 4 requires an upper bound on S, whereas PPE is adaptive to it. Lastly, Du et al. (2019) assumed deterministic setting while we provide a generalization to near-determinism.

from the path elimination data. We accomplish this by recovering a partition of paths in $\Psi_{h-1} \times \mathcal{A}$ where two paths in the same partition set are said to be *merged* with each other. In the beginning, each path is only merged with itself. When we eliminate a path υ_j on comparison with υ_i in line 8, then all paths currently merged with υ_j get merged with υ_i. We then define an abstract state space $\widehat{\mathcal{S}}_h$ for time step h that contains an abstract state j for each path $\upsilon_j \in \Psi_h$. Further, we recover a latent deterministic transition dynamics for time step $h-1$ as $\hat{T}_{h-1} : \widehat{\mathcal{S}}_{h-1} \times \mathcal{A} \to \widehat{\mathcal{S}}_h$ where we set $\hat{T}_{h-1}(i, a) = j$ if the path $\upsilon_j \in \Psi_h$ gets merged with path $\upsilon'_i \circ a \in \Psi_h$ where $\upsilon'_i \in \Psi_{h-1}$.

Learning a near optimal policy given a policy cover. `PPE` runs in a reward-free setting. However, the recovered policy cover and dynamics can be directly used to optimize any given reward function with existing methods. If the reward function depends on the exogenous state then we can use the `PSDP` algorithm (Bagnell et al., 2004) to learn a near-optimal policy. `PSDP` is a model-free dynamic programming method that only requires policy cover as input (see Appendix D.1 for details). If the reward function only depends on the endogenous state, we can use a computationally cheaper value-iteration `VI` that uses the recovered transition dynamics. `VI` is a model-based algorithm that estimates the reward for each state and action, and performs dynamic programming on the model (see Appendix D.2 for details). In each case, the sample complexity of learning a near-optimal policy, given the output of `PPE`, scales with the size of endogenous and not the exogenous state space.

5 Theoretical Analysis and Discussion

We provide the main sample complexity guarantee for `PPE` as well as additional intuition for why it works. We analyze the algorithm in near-deterministic MDPs defined as follows: Two transition functions T_1 and T_2 are *η-close* if for all $h \in [H], a \in \mathcal{A}, s \in \mathcal{S}_h$ it holds that $||T_1(\cdot \mid s, a) - T_2(\cdot \mid s, a)||_1 \leq \eta$. Analogously, two starting distribution μ_1 and μ_2 are η-close if $||\mu_1(\cdot) - \mu_2(\cdot)||_1 \leq \eta$. We emphasize that near-deterministic dynamics are common in real-world applications like robotics.

Assumption 1 (Near deterministic endogenous dynamics)**.** *We assume the endogenous dynamics is η-close to a deterministic model $(\mu_{D,\eta}, T_{D,\eta})$ where $\eta \leq 1/(4SH)$.*

We make a realizability assumption for the regression problem solved by `PPE` (line 5). We assume that $\mathcal{F}$ is expressive enough to represent the Bayes optimal classifier of the regression problems created by `PPE`.

Assumption 2 (Realizability)**.** *For any $h \in [H]$, and any set of paths $\Upsilon \subseteq \mathcal{A}^h$ with $|\Upsilon| \leq SA$ and where $\mathcal{A}^h$ denotes the set of all paths of length h, there exists $f^\star_{\Upsilon,h} \in \mathcal{F}$ such that: $f^\star_{\Upsilon,h}(\mathtt{idx}(\upsilon) \mid x) = \frac{\mathbb{P}_h(\phi^\star(x))|\upsilon)}{\sum_{\upsilon' \in \Upsilon} \mathbb{P}_h(\phi^\star(x))|\upsilon')}$, for all $\upsilon \in \Upsilon$ and $x \in \mathcal{X}$ with $\sum_{\upsilon' \in \Upsilon} \mathbb{P}_h(\phi^\star(x)) \mid \upsilon') > 0$.*

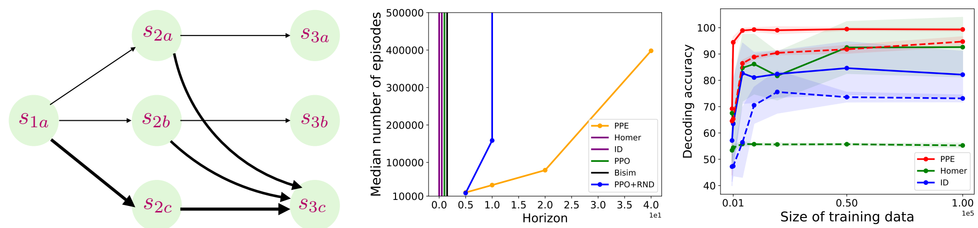
Realizability assumptions are common in theoretical analysis (e.g., Misra et al. (2020), Agarwal et al. (2020a)). In practice, we use expressive neural networks to solve the regression problem, so we expect the realizability assumption to hold. Note that there are at most $A^{S(H+1)}$ Bayes classifiers for different prediction problems. However, this is acceptable since our guarantees will scale as $\ln |\mathcal{F}|$ and, therefore, the function class $\mathcal{F}$ can be exponentially large to accommodate all of them.

We now state the formal sample complexity guarantees for `PPE` below.

Theorem 1 (Sample Complexity)**.** *Fix $\delta \in (0, 1)$. Then, with probability greater than $1 - \delta$,* `PPE` *returns a policy cover $\{\Psi_h\}_{h=1}^H$ such that for any $h \in [H]$, Ψ_h is a ηH-policy cover for time step h and $|\Psi_h| \leq S$, which gives the total number of episodes used by* `PPE` *as $\mathcal{O}\left(S^2 A^2 H \ln \frac{|\mathcal{F}|SAH}{\delta}\right)$.*

We defer the proof to Appendix C. Our sample complexity guarantees do not depend directly on the size of observation space or the exogenous space. Further, since our analysis only uses standard uniform convergence arguments, it extends straightforwardly to infinitely large function classes by replacing $\ln |\mathcal{F}|$ with other suitable complexity measures such as Rademacher complexity.

Why does `PPE` work? We provide an asymptotic analysis to explain why `PPE` works. Consider a deterministic setting and the h^{th} iteration of `PPE`. Assume by induction that Ψ_{h-1} is an exact policy cover for time step $h-1$. Therefore, $\Upsilon_h = \Psi_{h-1} \circ \mathcal{A}$ is also a policy cover for time step h. However, it may contain redundancies; it may contain several paths that reach the same endogenous state. We now show how a generalized inverse dynamics objective can eliminate such redundant paths.



(a) Combination lock ($H = 2$).

(b) Regret plot

(c) Decoding accuracy

Figure 2: Results on combination lock. **Left:** We show the latent transition dynamics of combination lock. Observations are not shown for brevity. **Center:** Shows minimal number of episodes needed to achieve a mean regret of at most $V(\pi^\star)/2$. **Right:** State decoding accuracy (in percent) of decoders learned by different methods. Solid lines implies no exogenous dimension while dashed lines imply an exogenous dimension of 100.

Let $\mathbb{P}_h(\xi)$ denote the distribution over exogenous states at time step h which is independent of agent's policy. The Bayes optimal classifier ($f_h^\star := f_{\Upsilon_h,h}$) of the prediction problem can be derived as:

$$f_h^\star(\texttt{idx}(\upsilon) \mid x) := \mathbb{P}_h(\upsilon \mid x) = \frac{\mathbb{P}_h(x \mid \upsilon)\mathbb{P}(\upsilon)}{\sum_{\upsilon' \in \Upsilon_h} \mathbb{P}_h(x \mid \upsilon')\mathbb{P}(\upsilon')} \overset{(a)}{=} \frac{\mathbb{P}_h(x \mid \upsilon)}{\sum_{\upsilon' \in \Upsilon_h} \mathbb{P}_h(x \mid \upsilon')} \overset{(b)}{=} \frac{\mathbb{P}_h(\phi^\star(x)) \mid \upsilon)}{\sum_{\upsilon' \in \Upsilon_h} \mathbb{P}_h(\phi^\star(x)) \mid \upsilon')},$$

where (a) holds since all paths in Υ_h are chosen uniformly, and (b) critically uses the fact that for any open-loop policy υ we have a factorization property,

$$\mathbb{P}_h(x \mid \upsilon) = q\left(x \mid \phi^\star(x), \phi_\xi^\star(x)\right) \mathbb{P}_h(\phi^\star(x) \mid \upsilon)\mathbb{P}_h(\phi_\xi^\star(x)).$$

Let $\upsilon_1, \upsilon_2 \in \Upsilon_h$ be two paths with indices i and j respectively. We define their exact path prediction gap as $\Delta(i,j) := \mathbb{E}_{x_h}\left[|f_h^\star(i \mid x_h) - f_h^\star(j \mid x_h)|\right]$. Assume that υ_1 visits an endogenous state s at time step h and denote $\omega(s)$ as the number of paths in Υ_h that reaches s. Then $f_h^\star(i \mid x_h) = 1/\omega(s)$ if $\phi^\star(x_h) = s$, and 0 otherwise. If υ_2 also visits s at time step h, then $f_h^\star(i \mid x_h) = f_h^\star(j \mid x_h)$ for all x_h. This implies $\Delta(i,j) = 0$ and PPE will filter out the path with higher index since it detected both paths reach to the same endogenous state. Conversely, let υ_2 visit a different state at time step h. If x is an observation that maps to s, then $f_h^\star(i \mid x) = 1/\omega(s)$ and $f_h^\star(j \mid x) = 0$. This gives $|f_h^\star(i \mid x) - f_h^\star(j \mid x)| = 1/\omega(s) \geq 1/|\Upsilon_h|$ and, consequently, $\Delta(i,j) > 0$. In fact, we can show $\Delta(i,j) \geq \mathcal{O}(1/|\Upsilon_h|)$. Thus, PPE will not eliminate these paths upon comparison. Our complete analysis in the Appendix generalizes the above reasoning to finite sample setting where we can only approximate $f_h^\star$ and Δ, as well as to EX-BMDPs with near-deterministic dynamics.

As evident, the analysis critically relies on the factorization property that holds for open-loop policies but not for arbitrary ones. This is the reason why we build a policy cover with open-loop policies.

6 Experiments

We evaluate PPE on two domains: a challenging exploration problem called *combination lock* to test whether PPE can learn an optimal policy and an accurate state decoder, and a visual-grid world with complex visual representations to test whether PPE is able to recover the latent dynamics.

Combination Lock Experiments. The combination lock problem is defined for a given horizon H by an endogenous state space $\mathcal{S} = \{s_{1,a}\} \cup \{s_{h,a}, s_{h,b}, s_{h,c}\}_{h=2}^H$, an exogenous state space $\Xi = \{0,1\}^H$, an action space $\mathcal{A}$ with 10 actions, and a deterministic endogenous start state of $s_{1,a}$. For any state $s_{h,g}$ we call g as its *type* which can be a, b or c. States with type a and b are considered *good* states and those with type c are considered *bad* states. Each instance of this problem is defined by two good action sequences $(a_h)_{h=2}^H, (a'_h)_{h=2}^H$ with $a_h \neq a'_h$, which are chosen uniformly randomly and kept fixed throughout. At $h = 1$, the agent is in $s_{1,a}$ and action a_1 leads to $s_{2,a}$, a'_h leads to $s_{2,b}$, and all other actions lead to $s_{2,c}$. For $h > 2$, taking action a_h in $s_{h,a}$ leads to $s_{h+1,a}$ and taking action a'_h in $s_{h,b}$ leads to $s_{h+1,b}$. In all other cases involving taking an action in a state $s_{h,g}$, we transition to the next bad state $s_{h+1,c}$. We visualize the latent endogenous dynamics in Figure 2a. The exogenous state evolves as follows. We set $\xi_1 \in \{0,1\}^H$ where $\xi_1(i) \sim \texttt{Unf}(\{0,1\})$ for each $i \in [H]$. At time step h, ξ_h is generated from ξ_{h-1} by uniformly flipping each bit in ξ_{h-1} independently with probability 0.1. There is a reward of 1.0 on taking the good action $a_{H,a}$ in $s_{H,a}$ and a reward of 0.1 on taking action $a_{H,b}$ in $s_{H,b}$. Otherwise, the agent gets a reward of 0. This gives a $V(\pi^\star) = 1$, and the probability that a random open loop policy gets this optimal return is 10^{-H}.

An observation x is generated stochastically from a latent state $z = (s, \xi)$. We map s to a vector w encoding the identity of the state. We concatenate (w, ξ), add Gaussian noise to each dimension, and multiply the result by a Hadamard matrix to generate x. See Appendix F for full details. Our construction is inspired by prior work (Du et al., 2019; Misra et al., 2020).

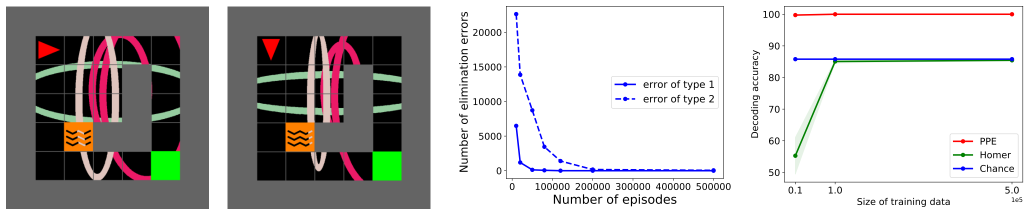
Baseline. We compare PPE with five baselines on the combination lock problem. These include PPO (Schulman et al., 2017) which is an actor-critic algorithm, PPO + RND (Burda et al., 2019) which adds an exploration bonus to PPO using prediction errors, Homer that uses contrastive learning (Misra et al., 2020), and another algorithm ID which is similar to Homer but instead of contrastive learning it learns an inverse dynamics model to recover the state abstraction. Lastly, we also compare with Bisim that learns a bisimulation metric along with an actor-critic agent (Zhang et al. (2020)). We use existing publicly available codebases for these baselines. Our implementation of PPE very closely follows the pseudo-code in Algorithm 1. We model $\mathcal{F}$ using a two-layer feed-forward network with ReLU non-linearity. We train $\mathcal{F}$ with Adam optimization and use a validation set to do model selection. We refer readers to Appendix F for additional experimental details.

Results. Figure 2b shows results for values of H in $\{5, 10, 20, 40\}$. For each value of H, we plot the minimal number of episodes n needed to achieve a mean regret of at most $V(\pi^\star)/2 = 0.5$. We run each algorithm 5 times with different seeds and report the median performance. If an algorithm is unable to achieve the desired regret in 5×10^5 episodes we set $n = \infty$. We observe that PPO is unable to solve the problem at $H = 5$. PPO + RND is able to solve the problem at $H = 5$ and $H = 10$, showing the exploration bonus induced by random network distillation helps. However, it is unable to solve the problem for larger values of H. We observe that Homer and ID are also unable to solve the problem for any value of H. Bisim also fails to solve the problem for any $H \geq 5$. This agrees with the theoretical prediction that Bisim provides no learning signal when running in sparse-reward settings. In the absence of any reward, the bisimulation objective incentivizes mapping all observations to the same representation which is not helpful for further exploration. Lastly, PPE is able to solve the problem for all values of H and is significantly more sample efficient than baselines. Since the reward function of the combination-lock problem depends only on the endogenous state, we run PPE and then a value-iteration like algorithm (see Appendix D.2) to learn a near optimal policy.

In order to understand the failure of Homer and ID, we investigate the accuracy of the state abstraction learned by these methods and compare that with PPE. We focus on the combination lock setting with $H = 2$ and evaluate the learned decoder for the last time step. As the state abstraction models are invariant to label permutation we use the following evaluation metric: given a learned abstraction for the endogenous state $\hat{\phi} : \mathcal{X} \to [N]$ we compute $1/m \sum_{i=1}^{m} \mathbf{1}\{\hat{\phi}(x_{i,1}) = \hat{\phi}(x_{i,2}) \Leftrightarrow \phi^\star(x_{i,1}) = \phi^\star(x_{i,2})\}$, where $\{x_{i,1}, x_{i,2}\}_{i=1}^{m}$ are drawn independently from a fixed distribution D with good support over all states. We report the percentage accuracy in Figure 2c. When there is no exogenous noise, Homer is able to learn a good state decoder with enough samples while ID fails to learn, in accordance with the theory. On inspection, we found that ID suffers from the under-abstraction issue highlighted earlier as it has difficulty separating observations from s_{3a} and s_{3b}. On adding exogenous noise, the accuracy of Homer plummets significantly. The accuracy of ID also drops but this drop is mild since unlike Homer, the ID objective is able to filter exogenous noise. Lastly, we observe that PPE is always able to learn a good decoder and is more sample efficient than baselines.

Visual Grid World Experiments. We test the ability of PPE to recover the latent endogenous transition dynamics in visual grid-world problem.[2] The agent navigates in a $N \times N$ grid world where each grid can contain a stationary object, the goal, or the agent. The agent's endogenous state is given by its position in the grid and its direction amongst four possible canonical directions. The agent can take five different actions for navigation. The world is visible to the agent as a $8N \times 8N$ sized RGB image. We add exogenous noise as follows: at the beginning of each episode, we independently sample position, size and color of 5 ellipses. The position and size of these ellipses is perturbed after each time step independent of the action. We project these ellipses on top of the world's image. Figure 3 shows sampled observations from the 7×7 gridworld that we experiment on. The

[2]We use the following popular gridworld codebase: https://github.com/maximecb/gym-minigrid



(a) Grid World $h = 1$. (b) Grid World $h = 2$. (c) Model Errors (d) State decoding accuracy

Figure 3: Results on visual grid world. **Left two:** Shows sampled observations for the first two steps from the visual gridworld domain. The agent is depicted as a red-triangle, lava in orange, walls in grey, and the goal in green. **Center Right:** Shows errors of type 1 and type 2 made by the PPE in recovering the latent endogenous dynamics. **Right:** State decoding accuracy of PPE, Homer and a random uniform decoder. (see Section 6)

exogenous state is given by the position, size and color of ellipses and is much larger than $|\mathcal{S}| \leq 4N^2$. We model $\mathcal{F}$ using a two-layer convolutional neural network and train it using Adam optimization. We defer the full details of setup to Appendix F.

Since the problem has deterministic dynamics, we evaluate the accuracy of the learned transition model by measuring it in terms of accuracy of the elimination step (Algorithm 1, line 8), since this step induces our algorithm's mapping from observations to endogenous latent states. For a fixed $h \in \{2, \cdots, H\}$, let ν_i and ν_j be two paths in $\Psi_{h-1} \circ \mathcal{A}$. We compute two type of errors. Type 1 error computes whether PPE merged these paths, i.e., predicted them as mapping to the same abstract state, when they go to *different endogenous* states. Type 2 error computes whether PPE predicted the paths as mapping to different abstract states, when they map to the *same endogenous* state. We report the total number of errors of both types by summing over all values of h and all pairs of different paths in $\Psi_{h-1} \circ \mathcal{A}$. Type 1 errors are more harmful, since they can lead to exploration failure. Specifically, merging paths going to different states may result in the algorithm avoiding one of the two states when exploring at the next time step. Type 2 errors are less serious but lead to inefficiency due to using redundant paths for exploration.

Results. We report results on learning the model in in Figure 3c. We see that PPE is able to reduce the number of type 1 errors down to 0 using 2×10^5 episodes per time step. This is important since even a single type 1 error can cause exploration failures. Similarly, PPE is able to reduce type 2 errors and is able to get them down to 56 with 5×10^5 episodes. This is acceptable since type 2 errors do not cause exploration failures but only cause redundancy. Therefore, at 2×10^5 samples, the algorithm makes 0 type 1 errors and just a handful type 2 errors. This is remarkable considering that PPE compares roughly 2×10^5 pairs of paths in the entire run. Hence, it makes only $\leq 0.03\%$ type 2 errors. Further, the agent is able to plan using the learned transition model and receive the optimal return. We also evaluate the accuracy of state decoding on this problem. We compare the state decoding accuracy of PPE and Homer at $H = 2$ using an identical evaluation setup to the one we used for combination lock. Figure 3d shows the results. As expected, PPE rapidly learns a highly accurate decoder while Homer performs only as well as a random uniform decoder.

7 Conclusion

In this work, we introduce the EX-BMDP setting, an RL setting that models exogenous noise, ubiquitous in many real-world systems. We show that many existing RL algorithms fail in the presence of exogenous noise. We present PPE that learns a multi-step inverse dynamics to filter exogenous noise and successfully explores. We derive theoretical guarantees for PPE in near-deterministic setting and provide encouraging experimental evidence in support of our arguments. To our knowledge, this is the first such algorithm with guarantees for settings with exogenous noise. Our work also raises interesting future questions such as how to address the general setting with stochastic transitions, or handle more complex endogenous state representations. Another interesting line of future work direction is the analysis of other approaches that learn multi-step inverse dynamics (Gregor et al., 2016; Paster et al., 2020) and understanding whether these approaches can also provably solve EX-BMDPs.

Acknowlegments

We would like to thank the reviewers for their suggestions and comments. We acknowledge the help of Microsoft's GCR team for helping with the compute. YE is partially supported by the Viterbi scholarship, Technion.

References

Alekh Agarwal, Daniel Hsu, Satyen Kale, John Langford, Lihong Li, and Robert E Schapire. Taming the monster: A fast and simple algorithm for contextual bandits. In *International Conference on Machine Learning*, 2014.

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*, 2020a.

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. Optimality and approximation with policy gradient methods in markov decision processes. In *Conference on Learning Theory*, 2020b.

András Antos, Csaba Szepesvári, and Rémi Munos. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 2008.

J Andrew Bagnell, Sham M Kakade, Jeff G Schneider, and Andrew Y Ng. Policy search by dynamic programming. In *Advances in Neural Information Processing Systems*, 2004.

Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A Efros. Large-scale study of curiosity-driven learning. In *International Conference on Learning Representations*, 2018.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *International Conference on Learning Representations*, 2019.

Christoph Dann, Tor Lattimore, and Emma Brunskill. Unifying PAC and regret: Uniform PAC bounds for episodic reinforcement learning. In *Advances in Neural Information Processing Systems*, 2017.

Christoph Dann, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. On oracle-efficient PAC RL with rich observations. In *Advances in Neural Information Processing Systems*, 2018.

Thomas G Dietterich, George Trimponias, and Zhitang Chen. Discovering and removing exogenous state variables and rewards for reinforcement learning. *arXiv preprint arXiv:1806.01584*, 2018.

Simon S Du, Akshay Krishnamurthy, Nan Jiang, Alekh Agarwal, Miroslav Dudík, and John Langford. Provably efficient RL with rich observations via latent state decoding. In *International Conference on Machine Learning*, 2019.

Yonathan Efroni, Nadav Merlis, and Shie Mannor. Reinforcement learning with trajectory feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

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

Robert Givan, Thomas Dean, and Matthew Greig. Equivalence notions and model minimization in markov decision processes. *Artificial Intelligence*, 2003.

Karol Gregor, Danilo Jimenez Rezende, and Daan Wierstra. Variational intrinsic control. *arXiv preprint arXiv:1611.07507*, 2016.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning*, 2019.

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

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

John Langford and Tong Zhang. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in Neural Information Processing Systems*, 2008.

Michael Laskin, Aravind Srinivas, and Pieter Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning*. PMLR, 2020.

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

Aditya Modi, 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*. PMLR, 2020.

Keiran Paster, Sheila A McIlraith, and Jimmy Ba. Planning from pixels using inverse dynamics models. In *International Conference on Learning Representations*, 2020.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, 2017.

Aviv Rosenberg and Yishay Mansour. Online convex optimization in adversarial markov decision processes. In *International Conference on Machine Learning*, 2019.

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

Lior Shani, Yonathan Efroni, and Shie Mannor. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. #Exploration: A study of count-based exploration for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, 2017.

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

Amy Zhang, Rowan Thomas McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=-2FCwDKRREu`.

APPENDIX

We present the main notations in Table 1 on page 12. The rest of the Appendix is organized as follows:

1. In Appendix A we establish failure cases of several popular approaches for learning an optimal policy in the EX-BMDP model.
2. In Appendix B we study structural properties of an EX-BMDP which highlight some of its key features.
3. In Appendix C we prove Theorem 1 our main performance guarantee on PPE. There, we show that PPE returns an approximate policy cover with cardinality that is bounded by the size of the endogenous state space. Furthermore, the sample complexity of PPE does not depend on the cardinality of the exogenous state space at all.
4. In Appendix D we analyze two planning approaches that utilize the output of PPE to find a near optimal policy when there exists an access to a reward function. We consider both the cases that the reward function is general, and a reward function that depends on the endogenous state.
5. In Appendix E we supply with several existing results that are used throughout the analysis.
6. In Appendix F we describe in further detalis the experimental setting and supply with additional experiments.

Notation	**Meaning**
$\mathcal{X}$	Countable observation space. Potentially infinite
$\mathcal{S}$	Finite endogenous state space. Assumed to be finite
Ξ	Countable exogenous state space. Potentially infinite
$\mathcal{Z}$	State space given by $\mathcal{Z} = \mathcal{S} \times \Xi$. Potentially infinite.
$\mathcal{A}$	Finite action space
q	Emission function $\mathcal{S} \to \Delta(\mathcal{X})$
T	Transition function $\mathcal{X} \times \mathcal{A} \to \Delta(\mathcal{X})$
H	Horizon
$\phi_z^\star$	Maps observation to latent state $\mathcal{X} \to \mathcal{Z}$
$\phi^\star$	Maps observation to endogenous state $\mathcal{X} \to \mathcal{S}$
$\phi_\xi^\star$	Maps observation to exogenous state $\mathcal{X} \to \Xi$
μ	start state distribution (overload $\mu(x) = q(x \mid \phi_z^\star(x))\mu(\phi_z^\star(x))$).
Π	Finite policy class
$\mathcal{F}$	Finite function class $\mathcal{X} \times [N] \to \mathbb{R}$
h	Indicates a time step $h \in [H]$
$\mathcal{M}_D$	An η close deterministic MDP of the endogenous dynamics
T_D	Transition function of the η close deterministic MDP of the endogenous dynamics
T_ξ	Transition function of the exogenous state space
Ψ_h	A policy cover of h step policies
$\pi(1:h)$	The first h step policies of a non-stationary policy of length greater than h
π_h	The policy at the h^{th} time step of a non-stationary policy
$\Psi_h \circ \mathcal{A}$	Extended policy cover of Ψ_h, $\Psi_h \circ \mathcal{A} = \{\pi \textit{ an } h+1 \textit{ step policy } : \pi(1:h) \in \Psi_h, \pi_h \in \mathcal{A}\}$

Table 1: Notations used in the paper. We start indexing from 1. An episode is given by $(s_1, a_1, s_2, a_2, \cdots, s_H, a_H, s_{H+1})$

A FAILURE OF EXISTING APPROACHES IN THE PRESENCE OF EXOGENOUS NOISE

A.1 FAILURE OF CONSTRASTIVE LEARNING

In this section we formally show that the objective defined in Misra et al. (2020) separates states according to the exogenous part. That is, states that share the same endogenous state, but have different exogenous state will be separated by the Backward Kinematic Inseparability criterion (BKI) on which the objective of HOMER relies upon (see Misra et al. (2020), Definition 3). Thus, the abstraction learned by HOMER will have the cardinality of $|\mathcal{S}| \times |\Xi|$ and will scale with the number of exogenous states.

Recall the definition of BKI given in Misra et al. (2020).

Definition 3 (Backward Kinematic Inseparability)**.** *Two states $(s'_1, \xi'_1), (s'_2, \xi'_2) \in \mathcal{Z}$ are backward kinematically inseparable if for all distributions $u \in \Delta(\mathcal{Z} \times \mathcal{A})$ supported on $\mathcal{Z} \times \mathcal{A}$ and for all $z = (s, \xi) \in \mathcal{Z}, a \in \mathcal{A}$ we have*

$$\frac{T(s'_1, \xi'_1 | s, \xi, a) u(s, \xi, a)}{\sum_{\tilde{s}, \tilde{\xi}, \tilde{a}} T(s'_1, \xi'_1 | \bar{s}, \bar{\xi}, \bar{a}) u(\bar{s}, \bar{\xi}, \bar{a})} = \frac{T(s'_2, \xi'_2 | s, \xi, a) u(s, \xi, a)}{\sum_{\tilde{s}, \tilde{\xi}, \tilde{a}} T(s'_1, \xi'_1 | \bar{s}, \bar{\xi}, \bar{a}) u(\bar{s}, \bar{\xi}, \bar{a})}.$$

The BKI criterion unifies states if they cannot be differentiated w.r.t. any sampling distribution u over the previous time step.

Claim 1. *BKI can splits states with similar endogenous states*

This claim is quite generic as we show below - for a very generic class of MDPs BKI splits states with similar endogenous state. Specifically, this occurs when the exogenous process is deterministic.

Proof. Consider any endogenous dynamics and a deterministic exogenous dynamics, i.e., $T(\xi'|\xi) = \mathbb{1}\{\xi' = \xi\}$. Let $z'_1 = (s', \xi'_1)$ and $z'_2 = (s', \xi'_2)$ be two states with similar endogenous dynamics and different exogenous dynamics. Then, BKI splits z'_1 and z'_2, i.e., it treats them as separate states.

Indeed, in this case, by the EX-BMDP model assumptions, it holds that

$$\begin{aligned}
&\frac{T(s', \xi'_1 | s, \xi, a) u(s, \xi, a)}{\sum_{\tilde{s}, \tilde{\xi}, \tilde{a}} T(s', \xi'_1 | \bar{s}, \bar{\xi}, \bar{a}) u(\bar{s}, \bar{\xi}, \bar{a})} = \frac{T(s'_1 | s, a)\, \mathbb{1}\{\xi'_1 = \xi\}\, u(s, \xi, a)}{\sum_{\tilde{s}, \tilde{\xi}, \tilde{a}} T(s' | \bar{s}, \bar{a})\, \mathbb{1}\{\xi'_1 = \xi\}\, u(\bar{s}, \bar{\xi}, \bar{a})} \\
&\neq \frac{T(s', \xi'_2 | s, \xi, a) u(s, \xi, a)}{\sum_{\tilde{s}, \tilde{\xi}, \tilde{a}} T(s', \xi'_2 | \bar{s}, \bar{\xi}, \bar{a}) u(\bar{s}, \bar{\xi}, \bar{a})} = \frac{T(s' | s, a)\, \mathbb{1}\{\xi'_2 = \xi\}\, u(s, \xi, a)}{\sum_{\tilde{s}, \tilde{\xi}, \tilde{a}} T(s' | \bar{s}, \bar{a})\, \mathbb{1}\{\xi'_2 = \xi\}\, u(\bar{s}, \bar{\xi}, \bar{a})}
\end{aligned}$$

where the inequality holds since $\xi'_1 \neq \xi'_2$ (o.w. $z'_1 = z'_2$). □

A.2 FAILURE OF BISIMULATION METRIC

Recall the definition of the bisimulation relations Givan et al. (2003) (also given in Zhang et al. (2020), Definition 1).

Definition 4 (Bisimulation Relations)**.** *Given an MDP $\mathcal{M}$, an equivalence relation B between states is a bisimulation relation if for all states $s_i, s_j \in \mathcal{S}$ that are equivalent under B it holds that*

1. $r(s_i, a) = r(s_j, a) \quad \forall a \in \mathcal{A}$,
2. $\mathbb{P}(G|s_i, a) = \mathbb{P}(G|s_j, a) \quad \forall a \in \mathcal{A}, G \in \mathcal{S}_B$

where $\mathcal{S}_B$ is the partition of $\mathcal{S}$ under the relation B and $\mathbb{P}(G|s, a) = \sum_{s'} T(s'|s, a)$.

Let d be some distribution over $\mathcal{S}$. We say that an equivalence relation B is a d-restricted bisimulation relation if Definition 4 holds for all $s \in \mathcal{S}$ such that $d(s) > 0$. That is, if B is a bisimulation for all states in the support of d. Indeed, we have no information on states that are not in the support of d. For this reason, we cannot obtain any information on these states.

Claim 2 (With no reward function bismulation may unify all states)**.** *Assume that $r(s, a) = 0$ for all states $s \in \mathcal{S}$ for which $d(s) > 0$, i.e., in the support of d. Then, the abstraction $\phi(s) = 1$ (i.e., merge all states into a single state) for all $s \in \mathcal{S}$ is a valid d-restricted bisumlation relation.*

Proof. We show this abstraction is a valid d-restricted relation.

1. For all states restricted to d, meaning $s \in \{s \in \mathcal{S} : d(s) > 0\}$ it holds that $r(s, a) = 0$ for all $a \in \mathcal{A}$. Thus, the first requirement of Definition 4 is satisfied.

Algorithm 2 Exact Inverse Dynamics for Deterministic Initial State

1: $\Psi_1 = \emptyset$
2: **for** $h = 1, \dots, H-1$ **do**
3: Set the sampling distribution μ to be $\texttt{Unf}(\Psi_h)^a$
4: Get $f_\star$ that exactly minimizes the loss

$$\mathbb{E}_{x\sim\mu, a\sim\texttt{Unf}(\mathcal{A}), x'\sim T(\cdot|x,a)}\left[\left(f(a|x,x') - \mathbb{P}_\mu(a|x,x')\right)^2\right]$$

5: Get a consistent ID abstraction (see Definition 5) $\phi : \mathcal{X} \to [N]$
6: Set the policy cover for next time as $\Psi_{h+1} = \{\pi_i^\star\}_{i=1}^N$ where $\pi_i^\star$ is a deterministic policy that maximizes the reaching probability to the event $\phi(x) = i$.

[a] If Ψ_h is an empty set then x is a fixed deterministic state

2. When all states are merges it holds that $G = \Omega$. Thus, $\mathbb{P}(G|s_i, a) = \mathbb{P}(G|s_j, a) = 1$, and the second requirement of Definition 4 is satisfied.

□

A.3 Failure of Inverse Dynamics

We describe a totally deterministic setting where Inverse Dynamics (ID) fails. We comment that the counter-example for ID supplied by Misra et al. (2020) has stochastic starting state. In this section, we show that even for a deterministic MDP with deterministic starting state ID fails. Specifically, we assume access to an exact solution $f_\star$ of a regression oracle that learns ID. Then, we show that a naive approach that uses $f_\star$ (see Algorithm 2) fails to return a policy cover.

We start by formally defining the ID abstraction[3].

Definition 5 (Inverse Dynamics Consistency)**.** *Two observations $x_1', x_2' \in \mathcal{X}$ are consistent under the ID and an initial distribution μ if $\forall x \in \mathcal{X}, a \in \mathcal{A}$ either*

1. $\mathbb{P}_\mu(a|x, x_1') = \mathbb{P}_\mu(a|x, x_2')$,

2. or either one of the following holds (i) $\mathbb{P}_\mu(x, x_1') = 0$, (ii) $\mathbb{P}_\mu(x, x_2'|\texttt{Unf}(\mathcal{A})) = 0$.

where $x \sim \mu$ and $x_1', x_2' \sim T(\cdot|x, a)$.

Relaying on this notion, we define an ID abstraction as follows.

Definition 6 (Inverse Dynamics Abstraction)**.** *We say that an abstraction $\phi : \mathcal{X} \to [N]$ is an* ID abstraction *if for all $x_1', x_2' \in \mathcal{X}$ for which $\phi(x_1') = \phi(x_2')$ it holds that x_1', x_2' are consistent under the ID according to Definition 5.*

Before addressing the problem that arises relying on the ID abstraction we elaborate on this definition, and specifically part two of its. We claim that this part is necessary to make the ID abstraction well defined. We motivate this definition by the two following arguments.

First, the ID object is a conditional probability function, $\mathbb{P}_\mu(a|x, x')$. Thus, for $\mathbb{P}_\mu(x, x') = 0$ the conditional probability is not well defined. Indeed, part two of Definition 5 is a possible solution to this issue – without it, the definition of the ID abstraction is not mathematically defined.

Second, a regression oracle that learns $\mathbb{P}_\mu(a|x, x')$ is not affected – i.e., has similar loss – for all pairs (x, x') for which $\mathbb{P}_\mu(x, x'|\texttt{Unf}(\mathcal{A})) = 0$. That is, the loss of any $f_\star(a|x, x')$ that approximates $P_\mu(a|x, x')$ is not affected by values of $f_\star(a|x, x')$ for which $\mathbb{P}_\mu(x, x'|\texttt{Unf}(\mathcal{A})) = 0$. Thus, the output of the regression oracle $f_\star$ can have arbitrary values on these pairs. This may result in $f_\star(a|x, x_1') =$

[3] There may be several abstractions which "agree" with the ID objective $\mathbb{P}_\mu(a|x, x')$. Thus, we define a notion of consistent ID abstraction.

$f_\star(a|x, x_2')$ for observations for which $\mathbb{P}_\mu(x, x_1'|\texttt{Unf}(\mathcal{A})) = 0$ or $\mathbb{P}_\mu(x, x_2'|\texttt{Unf}(\mathcal{A})) = 0$. Put it differently, when learning $\mathbb{P}_\mu(a|x, x')$ we cannot get any information outside the support of $\mathbb{P}_\mu(x, x'|\texttt{Unf}(\mathcal{A})) = 0$ and, thus, the values for these (x, x') pairs can be arbitrary.

We say two observations x_1', x_2' have *no shared common parent* if for any x such that $\mathbb{P}_\mu(x, x_1'|\texttt{Unf}(\mathcal{A})) > 0$ it $\mathbb{P}_\mu(x, x_2'|\texttt{Unf}(\mathcal{A})) = 0$ and vice-versa. Our counter example relies on the following observation.

Claim 3 (Inverse Dynamics may Merge Observations with no Shared Parent)**.** *If two observations x_1', x_2' have no shared common parent then merging these states is always a consistent ID abstraction according to Definition 5.*

This claim is a direct consequence of part two of Definition 5: if for all x it holds that either $\mathbb{P}_\mu(x, x_1'|\texttt{Unf}(\mathcal{A})) = 0$ or $\mathbb{P}_\mu(x, x_2'|\texttt{Unf}(\mathcal{A})) = 0$ then the two observations x_1', x_2' may always be merged while resulting in a consistent abstraction according to the ID abstraction.

With this observation at hand, we construct a simple deterministic MDP for which an ID abstraction merges states that should not be merged; in the sense that no deterministic policy can reach both states.

Proposition 1 (Failure of Inverse Dynamics)**.** *The exists a deterministic MDP such that Algorithm 2 does not return a policy cover on the states on the second time step.*

Proof. Consider the MDP in Figure 2, (a). At $h = 1$, a consistent ID abstraction must separate the states s_{2a}, s_{2b}, s_{2c} and s_{2c}. Thus,Algorithm 2 separates all states at $h = 1$. However, at $h = 2$, since s_{3a} and s_{3b} share no common parent, a consistent ID abstraction may merge these states (due to Claim 3).

Then, since our policy class contains only deterministic policies, we get that necessarily at the end of the $h = 2$ iteration, the policy cover that Algorithm 2 returns does not contain a policy that reaches either s_{3a} or s_{3b}. That holds since any deterministic policy that maximizes the reaching probability to $s_{3a} \cup s_{3b}$ will hit either s_{3a} or s_{3b}. Thus, one of these states will not be reached by the policy cover. $\square$

A.4 Bellman Rank Depends on the Exogenous State Cardinally

Proposition 2 (Bellman Rank Depends on the Exogenous State Cardinality)**.** *There exists an Exogenous Block MDP $\mathcal{M}$, policy class Π, and value function class $\mathcal{F}$ with the following properties: (1) the endogenous state has size 2, (2) the exogenous state has size d, (3) $|\Pi| = |\mathcal{F}| = O(d)$, (4) the optimal policy and value function are in $\Pi, \mathcal{F}$ respectively, and (5) the $(\mathcal{F}, \Pi)$ bellman rank is $\Omega(d)$. Additionally,* OLIVE *has sample complexity $\Omega(poly(d)/\epsilon^2)$ to learn an ϵ-optimal policy.*

Proof. We construct the Exogenous Block MDP as follows. Let the horizon be H and set the starting endogenous state to be labeled g_1. From g_1 there are two actions: action a_1 transits to g_2 while action a_2 transits to b_2. We repeat this $H-1$ times randomizing the good and bad action at each level, to arrive at either g_H or b_H. (Here g denotes "good" and b denotes "bad".) Let $a_h^\star$ denote the good action at time h and $\tilde{a}_h$ denote the bad action at time h.

There are no actions at time H and from g_H the agent always receives reward 1, while from b_H the agent always receives reward 0. There are no intermediate rewards. The exogenous state ξ does not change across time and, at the beginning of the episode, ξ is drawn uniformly from $[d]$.

The policy class Π is defined as $\Pi = \{\pi_\star, \pi_1, \ldots, \pi_d\}$ where $\pi_\star((g_h, \cdot)) = a_h^\star$ and π_i agrees with $\pi_\star$ everywhere, except for at state (g_{H-1}, i) where it takes $\tilde{a}_{H-1}$. In other words, π_i defects from the optimal policy only in the good state at time $H-1$ when the exogenous variable $\xi = i$. Meanwhile, the value function class is $\mathcal{F} = \{f_\star, f_1, \ldots, f_d\}$ where $f_\star$ is the optimal value function which satisfies

$$f_\star((g_h, \cdot)) = 1, \qquad f_\star((b_h, \cdot)) = 0$$

and f_i deviates from the $f_\star$ only on state (b_H, i), where $f_i((b_H, i)) = 1$.

First observe that (f_i, π_i) is clearly bellman consistent at all time steps except for the time step H, since f_i predicts 1 at all states up to time $H-1$ and on all states visited by π_i at time H.

Now, let us examine the bellman error on roll-in π_j for value function/policy pair (f_i, π_i) at the last time. First, the state distribution visited by π_j at time H is Unif$(\{g_H, k\}_{k \neq j} \cup \{b_H, j\})$. If $i \neq j$ then (f_i, π_i) has zero bellman error on all of these states since it correctly predicts that the reward is 1 in the good state and it correctly predicts that the reward is 0 on the bad state (b_H, j).

On the other hand, if $i = j$ then the bellman error is $1/d$, since f_j *incorrectly* predicts that the reward is 1 on the bad state. Thus we see that

$$\mathcal{E}_H(\pi_i, (g_j, \pi_j)) := \mathbb{E}_{(s_H, \xi_H, r_H) \sim \pi_i}\left[g_j(s_H, \xi_H) - r_H\right] = \frac{\mathbf{1}\{i = j\}}{d}.$$

This verifies that the bellman rank is d.

Regarding OLIVE, note that the value functions are all bellman consistent at the first $H-1$ time steps. In particular,

$$\forall h \le H-1 : \mathbb{E}_{\pi_i}[f_i((g_h, \xi)) - f_i((s_{h+1}, \xi))] = 0$$

Thus OLIVE is unable to eliminate functions using data at the first $H-1$ time steps. Additionally even with perfect evaluation of expectation, due to adversarial tie breaking, OLIVE may take $\Omega(d)$ iterations to find the optimal policy, since it may cycle through $\pi_1, \ldots, \pi_d$ eliminating one at a time.

This argument can be extended to get $poly(d)/\epsilon^2$ sample complexity using standard technique. To prove a lower bound for the sample complexity of `OLIVE` we adjust the problem by making the rewards $1/2$ and $1/2 + \epsilon$. Then, to eliminate a bad policy, we need to estimate the Bellman errors to accuracy of ϵ/d. This results in an $poly(d)/\epsilon^2$ lower bound for the sample complexity of `OLIVE`.

□

A.5 Failure of `FLAMBE` Agarwal et al. (2020a).

In Agarwal et al. (2020a) the authors studied a representation learning problem for the linear MDP setting and suggested an algorithm, `FLAMBE`, that provably explores while learning the representation feature map of the linear MDP model. Their algorithm relies on a model-based approach to factorize the transition dynamics. However, focusing on the dynamics in observation space forces the modeling of the exogenous state as well, and the dimension of the factorization that they learn can scale with $|\Xi|$ (similar to the Bellman rank), leading to a poly$(|\Xi|)$ sample complexity for their approach.

A.6 FAILURE OF AUTO-ENCODING APPROACHES

Much prior work uses auto-encoding or other unsupervised techniques for representation learning in RL. Examples include scalable count-based exploration methods (Tang et al., 2017) and CURL (Laskin et al., 2020). However, as these methods do not leverage the temporal nature of reinforcement learning, it is easy to see that the representations discovered may not be useful or relevant for exploration or policy learning, without relying heavily on inductive biases. More concretely, an autoencoding approach that aims to minimize reconstruction error on observations would prefer to memorize high-entropy irrelevant noise over lower-entropy relevant state information (see figure 4c in (Misra et al., 2020)). Unfortunately, the resulting learned representation may omit state information that is crucial for downstream planning.

B STRUCTURAL RESULTS FOR EX-BMDP

In this section we prove several useful structural results about the EX-BMDP model which will be essential for later analysis. A key definition which will be useful is the notion of *endogenous policy.* We define the class of *endogenous policies* Π_d as the policies that depend only on the *endogenous* part of the state. Formally, a policy $\pi \in \Pi_d$ has the property that for all $x \in \mathcal{X}: \;\; \pi(a|x) = \pi(a|\phi^\star(x))$. Restated, an endogenous policy chooses the same actions for a fixed endogenous state across varying exogenous states and observations. See that *open loop* policies (see definition in Section 2), which commit to a sequence of actions prior to the interaction, are always endogenous policies, since this sequence of actions is independent of the exogneous noise.

Proposition 3 (Consequence of Endogenous Policy)**.** *Let $\pi \in \Pi_d$. Then, for any $h \in [H]$ it holds that $\mathbb{P}_h(z|\pi) = \mathbb{P}_h(s|\pi)\mathbb{P}_h(\xi)$. Furthermore, there exists $\pi \in \Pi_{NS}\backslash\Pi_d$ such that $\mathbb{P}_h(z|\pi) \neq \mathbb{P}_h(s|\pi)\mathbb{P}_h(\xi)$.*

Proof. **First claim.** We prove the result by induction.
Base case $h = 1$**.** The base case follows from the model assumption, that is,

$$\mathbb{P}_{h=1}(z|\pi) = \mu(z) = \mu(s)\mu_\xi(\xi).$$

Induction step. Assume the claim holds for h. We show it also holds for $h+1$ for $\pi \in \Pi_d$. For any $z \in \mathcal{Z}$, or, equivalently $z = (s, \xi)$ for $s \in \mathcal{S}, \xi \in \Xi$, it holds that

$$\begin{aligned}
\mathbb{P}_{h+1}(z|\pi) &= \sum_{z_h \in \mathcal{Z}} \mathbb{P}_{h+1}(z, z_{h-1}|\pi) && \text{(Law of total probability)}\\
&= \sum_{z_h \in \mathcal{Z}} \mathbb{P}_h(z_h|\pi) T(z|z_h, \pi) && \text{(Bayes' theorem \& Markovian dynamics)}\\
&= \sum_{s \in \mathcal{S}, \xi \in \Xi} \mathbb{P}_h(s_h|\pi)\mathbb{P}_h(\xi_h) T(z|z_h, \pi) && \text{(Induction hypothesis)}\\
&= \sum_{s \in \mathcal{S}, \xi \in \Xi} \mathbb{P}_h(s_h|\pi)\mathbb{P}_h(\xi) T(s|s_h, \pi) T(\xi|\xi_h) && (\pi \in \Pi_d \text{ and EX-BMDP definition})\\
&= \sum_{s \in \mathcal{S}} \mathbb{P}_h(s_h|\pi) T(s|s_h, \pi) \sum_{\xi \in \Xi} \mathbb{P}_h(\xi) T(\xi|\xi_h)\\
&= \mathbb{P}_{h+1}(s|\pi)\mathbb{P}_{h+1}(\xi),
\end{aligned}$$

which concludes the proof.

Second claim. Consider an MDP with $H = 2$. The endogenous state is $b_d \in \{0, 1\}$, the exogenous state is $b_\xi \in \{0, 1\}$ and the full state is $z = \{b_d, b_\xi\}$. At the initial time step $b_d = 1$ and $b_\xi \sim \texttt{Unf}(\{0, 1\})$. Furthermore, the MDP is stationary, and its dynamics is given as follows. The exogenous state is fixed along trajectory,

$$T(b'_\xi = 1|b'_\xi = 1) = T(b'_\xi = 0|b'_\xi = 0) = 1.$$

The action set is of size two, $\mathcal{A} = \{0, 1\}$, the endogenous dynamics evolves as

$$T(b'_d = 1|b_d = 1, a = 0) = T(b'_d = 0|b_d = 1, a = 1) = 1$$

that is, applying $a = 1$ the endogenous state switches to 0 and applying $a = 0$ leaves the endogenous state at 1.

Let the policy be

$$\pi(a = 1|b_\xi) = \begin{cases} 1 & b_\xi = 1 \\ 0 & o.w. \end{cases},$$

and observe it depends on the exogenous state.

We show that the next state distribution does not decouple the endogenous and exogenous part of the state space.

$$\begin{aligned} &\mathbb{P}_{h=2}(z|\pi) = \mathbb{P}_{h=2}(b_\xi)\mathbb{P}_{h=2}(s|\pi, \xi) \\ &= 0.5\left(\mathbb{I}\left\{b_\xi = 1, s = 0\right\} + \mathbb{I}\left\{b_\xi = 0, s = 1\right\}\right) \\ &\neq \mathbb{P}_{h=2}(s|\pi)\mathbb{P}_{h=2}(b_\xi) = 0.25. \end{aligned}$$

□

Proposition 4 (Existence of Endogenous Policy Cover)**.** *Given an EX-BMDP, for any $h \in [H]$, let Ψ_h be an endogenous policy cover of the endogenous dynamics, that is $\Psi_h \subseteq \Pi_d$ and for all $s \in \mathcal{S}$, $\max_{\pi\in\Psi_h} \mathbb{P}_h(s|\pi) = \max_\pi \mathbb{P}_h(s|\pi)$. Then, Ψ_h is also a policy cover for the full EX-BMDP, that is: $\max_{\pi\in\Psi_h} \mathbb{P}_h(z|\pi) = \max_\pi \mathbb{P}_h(z|\pi)$.*

Proof. Let Π_{NS} be the set of tabular policies, $\pi \in \Pi_{NS}$ is a mapping $\pi : \mathcal{Z} \to \mathcal{A}$[4]. Fix $\bar{z} \in \mathcal{Z}_h$, where $\bar{z} = \left(\bar{s}, \bar{\xi}\right)$ for some $\bar{s}$ and $\bar{\xi}$ by the EX-BMDP model assumption and fix $h \in [H]$. We not show that for $\bar{z}$ there exists an optimal policy which is also endogenous policy that reaches $\bar{z}$. That is, we show that

$$\max_{\pi\in\Pi_{NS}} \mathbb{P}_h(\bar{z}|\pi) = \max_{\pi\in\Pi_d} \mathbb{P}_h(\bar{z}|\pi).$$

See that

$$\max_{\pi\in\Pi_{NS}} \mathbb{P}_h(\bar{z}|\pi) = \max_{\pi\in\Pi_{NS}} \mathbb{E}\left[\sum_{h'=1}^{h} r_{h'}(z_{h'}) \mid \pi\right],$$

where $r_h(z) = \mathbb{I}\left\{z = \bar{z}\right\}$ and zero for all other time steps. We now show inductively that the optimal Q function of the MDP $\mathcal{M}_{\bar{z}} = (\mathcal{Z}, \mathcal{A}, r, T, h)$ is given by

$$Q^\star_{h'}(z, a) = \mathbb{P}(\bar{\xi}_h|\xi_{h'})Q^\star_{d,h'}(s, a),$$

where $Q^\star_{d,h'}(s, a)$ does not depend on ξ where $\mathbb{P}(\bar{\xi}_h|\xi_{h'})$ is the probability the exogenous state at time step h is ξ_h given it is $\xi_{h'}$ at time step h'. Specifically, $Q^\star_{d,h'}(s, a)$ is the optimal Q function on the endogenous MDP $\mathcal{M}_{\bar{s}} = (\mathcal{S}, \mathcal{A}, r_d, T, h)$ with reward $r_{d,h}(s) = \mathbb{I}\left\{s = \bar{s}\right\}$ and zero for all other time steps, defined on the endogenous state space. This implies that the there exists an optimal policy of $\mathcal{M}_{\bar{z}}$ which is an endogenous policy. The policy

$$\pi^\star_{\bar{z}} \in \arg\max_a Q^\star_{d,h'}(s, a)$$

is an endogenous policy which is also optimal.

Base case $h' = h$**.** For the last time step, the claim trivially holds

$$\begin{aligned} &Q^\star_h(z, a) = \mathbb{I}\left\{z = \bar{z}\right\} = \mathbb{I}\left\{\xi = \bar{\xi}\right\}\mathbb{I}\left\{s = \bar{s}\right\} \\ &= \mathbb{P}(\bar{\xi}_h|\xi_h)\,\mathbb{I}\left\{s = \bar{s}\right\} = \mathbb{P}(\bar{\xi}_h|\xi_h)Q^\star_{d,h}(s, a). \end{aligned}$$

[4]Observe that the tabular policies contain an optimal policy for the general Block-MDP model.

Induction step. Assume the claim holds for $h'+1$ and prove it holds for $h'<h$. The optimal Q function satisfies the following relations for any $h' \in [h]$ and $z_{h'} \in \mathcal{Z}$, since $r_{h'}=0$ for any $h'<h$.

$$
\begin{aligned}
&Q^\star_{h'}(z_{h'},a) = \mathbb{E}\left[\max_{a'} Q^\star_{h'+1}(z_{h'+1},a')|z_{h'},a\right]\\
&= \mathbb{E}\left[\max_{a'} \mathbb{P}(\bar{\xi}_h|\xi_{h'+1}) Q^\star_{d,h'+1}(s_{h'+1},a')|z,a\right] && \text{(Induction step)}\\
&= \mathbb{E}\left[\mathbb{P}(\bar{\xi}_h|\xi_{h'+1}) \max_{a'} Q^\star_{d,h'+1}(s_{h'+1},a')|s_{h'},\xi_{h'},a\right]\\
&= \sum_{\xi_{h'+1}} \mathbb{P}(\bar{\xi}_h|\xi_{h'+1}) T(\xi_{h'+1}|\xi_{h'}) \sum_{s_{h'+1}} T(s_{h'+1}|s_{h'},a) \max_{a'} Q^\star_{d,h'+1}(s_{h'+1},a')\\
& && \text{(Transition assumption)}\\
&= \mathbb{P}(\bar{\xi}_h|\xi_{h'}) Q^\star_{d,h'}(s_{h'},a),
\end{aligned}
$$

where

$$
Q^\star_{d,h'}(s_{h'},a) = \sum_{s_{h'+1}} T(s_{h'+1}|s_{h'},a) \max_{a'} Q^\star_{d,h'+1}(s_{h'+1},a'),
$$

does not depend on the exogenous part of the state space by the induction hypothesis since $Q^\star_{d,h'+1}(s_{h'+1},a')$ does not depend on the exogenous part of the state space. Furthermore, $Q^\star_{d,h'}(s_{h'},a)$ satisfy the Bellman equations of $\mathcal{M}_{\bar{s}}$, and, thus, it is the optimal Q function of $\mathcal{M}_{\bar{s}}$.

This concludes the proof of the induction step.

□

Proposition 5. *Consider an EX-BMDP and assume its reward function depends only on the endogenous part of the state, i.e., for all $x \in \mathcal{X}, a \in \mathcal{A}$ and $h \in [H]$, $r_h(x,a) = r_h(\phi^\star(x),a)$. Then, the class of endogenous policies Π_d contains an optimal policy.*

Proof. Let $r_h(x,a) = r_h(\phi^\star(x),a)$ be a reward function that depends on the endogenous state. To establish the claim, it is sufficient to prove that for any $h \in [H], x \in \mathcal{X}, a \in \mathcal{A}$ it holds that $Q^\star_h(x,a) = Q^\star_h(\phi^\star(x),a)$, where $\{Q^\star_h\}_{h=1}^H$ is the optimal Q function. This implies that

$$
\pi^\star_h \in \arg\max_a Q^\star_h(\phi^\star(x),a),
$$

is an optimal policy. Furthermore, $\pi^\star_h$ is an endogenous policy, since it does not depend on the exogenous part of the state space.

We prove this claim by induction.
Base case, $h=H$. Holds trivially since the reward function depends only on the endogenous state for all x,a. Thus, for all $x \in \mathcal{X}, a \in \mathcal{A}$

$$
Q^\star_H(x,a) = r_H(x,a) = r_H(\phi^\star(x),a),
$$

by assumption.

Induction step, h. Assume the claim holds for any $h+1$ for $h<H$. We now show it holds for the h^{th} time step. The optimal Q function satisfies the following relations for all $x \in \mathcal{X}, a \in \mathcal{A}$

$$
\begin{aligned}
&Q^\star_h(x,a) = r_h(x,a) + \mathbb{E}\left[\max_a Q^\star_{h+1}(x',a)|x,a\right]\\
&= r_h(\phi^\star(x),a) + \mathbb{E}\left[\max_a Q^\star_{h+1}(\phi^\star(x'),a)|x,a\right] && \text{(Induction hypothesis \& reward assumption)}\\
&= r_h(\phi^\star(x),a) + \sum_{s',\xi'} T(s'|\phi^\star(x),a) T_\xi(\xi'|\phi^\star_\xi(x)) \max_a Q^\star_{h+1}(s',a)\\
&= r_h(\phi^\star(x),a) + \sum_{s'} T(s'|\phi^\star(x),a) \max_a Q^\star_{h+1}(s',a). && \left(\textstyle\sum_{\xi'} T(\xi'|\phi^\star_\xi(x))=1\right)
\end{aligned}
$$

Thus, for all $x \in \mathcal{X}, a \in \mathcal{A}$ it holds that $Q^\star_h(x,a) = Q^\star_h(\phi^\star(x),a)$, i.e., is a function of the endogenous state. □

C Sample Complexity of PPE

In this section, we present a proof showing that PPE learns a policy cover for nearly deterministic EX-BMDPs. More formally, to EX-BMDPS such that the endogenous dynamics is η close to a deterministic MDP (see Assumption 1 to quantification of η close model).

C.1 Inverse Dynamics Filters Exogenous State

We start by establishing the following result which sheds further intuition on the performance of PPE. In words, it says that the inverse dynamics objective filters exogenous noise and depends only on the endogenous part of the state.

Lemma 1 (Inverse Dynamics Filters Exogenous State)**.** *For any endogenous policy $\pi \in \Pi_d$ and for all $a \in \mathcal{A}, h \in [H]$ it holds that $\mathbb{P}_h(a, \phi^\star(x_{h-1})|x_h, \pi) = \mathbb{P}_h(a, \phi^\star(x_{h-1})|\phi^\star(x_h), \pi)$. If $\pi \in \Pi_{NS}\backslash\Pi_d$, $\mathbb{P}_h(a, \phi^\star(x_{h-1})|x_h^{(2)}, \pi) \neq \mathbb{P}_h(a, \phi^\star(x_{h-1})|x_h^{(2)}, \pi)$ for $\phi^\star(x_h^{(1)}) = \phi^\star(x_h^{(2)})$.*

Proof. **First statement.** Proven in Lemma 2 by explicitly applying Bayes' theorem and the decoupling property of the future state distribution of $\pi \in \Pi_d$ (see Proposition 3).

Second statement. Consider an MDP with $H = 2$. The endogenous state is $b_d \in \{0, 1\}$, the exogenous state is $b_\xi \in \{0, 1\}$ and the full state is $z = \{b_d, b_\xi\}$. At the initial time step $b_d = 1$ and $b_\xi \sim \texttt{Unf}(\{0, 1\})$. Furthermore, the MDP is stationary, and its dynamics is given as follows. The exogenous state is fixed along a trajectory,

$$T_\xi(b'_\xi = 1|b'_\xi = 1) = T_\xi(b'_\xi = 0|b'_\xi = 0) = 1.$$

The action set is of size two, $\mathcal{A} = \{0, 1\}$, the endogenous dynamics evolves as

$$T(b'_d = 1|b_d = 1, a = 1) = 1 - \alpha,\ T(b'_d = 1|b_d = 1, a = 0) = \alpha.$$

Assume the policy is a function of the exogenous state given as follows

$$\pi(a = 1|b_\xi) = \begin{cases} 1 & b_\xi = 1 \\ 0 & b_\xi = 0 \end{cases}.$$

To conclude the proof, let b'_d, b'_ξ denote the state at the second time step. We show that (suppressing the deterministic event $b_d = 1$ at the first time step)

$$\frac{\mathbb{P}(a = 1|b'_d = 1, b'_\xi = 1)}{\mathbb{P}(a = 1|b'_d = 1, b'_\xi = 0)} \neq 1,$$

thus, $\mathbb{P}(a = 1|b'_d = 1, b'_\xi = 1) \neq \mathbb{P}(a = 1|b'_d = 1, b'_\xi = 0)$.

To prove this, by Bayes' theorem,

$$\frac{\mathbb{P}(a = 1|b'_d = 1, b'_\xi = 1)}{\mathbb{P}(a = 1|b'_d = 1, b'_\xi = 0)} = \frac{\mathbb{P}(b'_d = 1, b'_\xi = 1|a = 1)}{\mathbb{P}(b'_d = 1, b'_\xi = 0|a = 1)} \frac{\mathbb{P}(b'_d = 1, b'_\xi = 0)}{\mathbb{P}(b'_d = 1, b'_\xi = 1)}. \tag{1}$$

Observe that $\mathbb{P}(b'_d = 1, b'_\xi = 0|a = 1) = \mathbb{P}(b'_d = 1, b'_\xi = 1|a = 1) = 0.5 \cdot \alpha$. Thus,

$$equation\ 1 = \frac{\mathbb{P}(b'_d = 1, b'_\xi = 0)}{\mathbb{P}(b'_d = 1, b'_\xi = 1)}.$$

See that

$$\mathbb{P}(b'_d = 1, b'_\xi = 0) = \mathbb{P}(b'_d = 1|b'_\xi = 0)\mathbb{P}(b'_\xi = 0) = 0.5\alpha$$
$$\mathbb{P}(b'_d = 1, b'_\xi = 1) = \mathbb{P}(b'_d = 1|b'_\xi = 1)\mathbb{P}(b'_\xi = 1) = 0.5(1 - \alpha).$$

Thus, for $a \notin \{0, 1, 0.5\}$ we get that

$$\frac{\mathbb{P}(a = 1|b'_d = 1, b'_\xi = 1)}{\mathbb{P}(a = 1|b'_d = 1, b'_\xi = 0)} = \frac{\alpha}{1 - \alpha} \neq 1.$$

□

The intuition which underlies the construction of the second statement goes as follows. When a policy acts according to the exogenous state information on the exogenous state at time step $h = 2$ may change the knowledge we have on the action taken at the previous time step $h = 1$.

Lemma 2 (Bayes Optimal Classifier)**.** *Assume that $\pi \in \Pi_d$ is an endogenous policy. Then, for every $x, x' \in \mathcal{X}_h, a \in \mathcal{A}$ we have:*

$$\mathbb{P}(a, \phi^\star(x) \mid x', \pi) = \frac{T(\phi^\star(x') \mid a, \phi^\star(x))\mathbb{P}(a, \phi^\star(x) \mid \pi)}{\sum_{s\in\mathcal{S}, a\in\mathcal{A}} T(\phi^\star(x') \mid a, s)\mathbb{P}(a, s \mid \pi)}.$$

Proof. We prove this result by applying Bayes' Theorem.

$$\mathbb{P}(a, \phi^\star(x) \mid x', \pi) = \frac{\mathbb{P}(x' \mid a, \phi^\star(x), \pi)\mathbb{P}(a, \phi^\star(x) \mid \pi)}{\sum_{s\in\mathcal{S}, a\in\mathcal{A}} \mathbb{P}(x' \mid a, s, \pi)\mathbb{P}(a, s \mid \pi)} \tag{2}$$

Observe that by the model assumption,

$$\mathbb{P}(x' \mid a, \phi^\star(x), \pi) = q(x'|\phi^\star(x'), \phi^\star_\xi(x'))\mathbb{P}(\phi^\star(x'), \phi^\star_\xi(x') \mid a, \phi^\star(x), \pi). \tag{3}$$

We focus on the second term. By the law of total probability and by applying Bayes' theorem,

$$\begin{aligned}
&\mathbb{P}(\phi^\star(x'), \phi^\star_\xi(x') \mid a, \phi^\star(x), \pi) = \sum_\xi \mathbb{P}(\phi^\star(x'), \phi^\star_\xi(x') \mid a, \phi^\star(x), \xi, \pi)\mathbb{P}(\xi \mid a, \phi^\star(x), \pi) \\
&\underset{(a)}{=} \sum_\xi \mathbb{P}(\phi^\star(x'), \phi^\star_\xi(x') \mid a, \phi^\star(x), \xi, \pi)\mathbb{P}(\xi) \\
&\underset{(b)}{=} \sum_\xi T(\phi^\star(x') \mid a, \phi^\star(x))T_\xi(\phi^\star_\xi(x') \mid \xi)\mathbb{P}(\xi) \\
&= T(\phi^\star(x') \mid a, \phi^\star(x)) \sum_\xi T_\xi(\phi^\star_\xi(x') \mid \xi)\mathbb{P}(\xi) \\
&= T(\phi^\star(x') \mid a, \phi^\star(x))\mathbb{P}(\phi^\star_\xi(x')).
\end{aligned} \tag{4}$$

where (a) holds since the policy $\pi \in \Pi_d$ followed by a fixed action a is an endogenous policy and $\mathbb{P}(\xi \mid a, \phi^\star(x), \pi) = \mathbb{P}(\xi)$ for an endogenous policy due to Proposition 3. The relation (b) holds due to the Markov property of the latent state and the decoupling of the transition model of an EX-BMDP.

Plugging equation 3 and equation 4 into equation 2 we get

$$\mathbb{P}(a, \phi^\star(x) \mid x', \pi) = \frac{T(\phi^\star(x') \mid a, \phi^\star(x))\mathbb{P}(a, \phi^\star(x) \mid \pi)}{\sum_{s\in\mathcal{S}, a\in\mathcal{A}} T(\phi^\star(x') \mid a, s)\mathbb{P}(a, s \mid \pi)}.$$

□

C.2 HIGHLEVEL ANALYSIS OVERVIEW OF THEOREM 1

Analysis overview, deterministic dynamics. Prior to addressing the near deterministic case, we consider the fully deterministic case. The analysis of this setting highlights the core ideas that are later utilized in the proof.

The idea which underlies the analysis of PPE is to prove, in an inductive manner, that at each time step $h \in 2, 3., ..H$ the policy cover at Ψ_{h-1} is a *minimal policy cover* of the endogenous state space. That is, there is a one-to-one correspondence between endogenous states $s \in \mathcal{S}_{h-1}$ and open-loop policies $\nu \in \Psi_{h-1}$ such that for any $s \in \mathcal{S}_{h-1}$ there exists a unique open-loop policy $\upsilon \in \Psi_{h-1}$ that reaches s.

The base case holds since the starting state is deterministic. Assume the claim holds for $h - 1$. Then, we need to show it holds for h. Since Ψ_{h-1} is a minimal policy cover for the endogenous dynamics by a compositionality property of deterministic environments (see Lemma 6), the set of open-loop policies $\Upsilon_h = \Psi_{h-1} \circ \mathcal{A}$ in which every policy in Ψ_{h-1} is extended by all actions, is a policy cover for the h^{th} time step. However, it may contain duplicates; two paths may reach to the same endogenous state.

Due to the above reasoning, by eliminating policies from Υ_h we can recover a minimal policy cover for the h^{th} time step and to establish to induction step. Let $\nu \in \Upsilon_h$ and $f_h^\star(\nu, x') \equiv \mathbb{P}_h(\nu|x', \texttt{Unf}(\Upsilon_h))$ be the inverse dynamics which predicts the probability ν was taken while observing x' and following the policy $\texttt{Unf}(\Upsilon_h)$. Due to the one-to-one correspondence between $\{(s,a)\}_{a\in\mathcal{A}, s\in\mathcal{S}_{h-1}}$ and Υ_h we expect this function to depend only on the endogenous state (see Proposition 4). Specifically, it is possible to show the following identity on which the elimination criterion of PPE is based upon. Let $\Delta^\star(i,j) = \mathbb{E}_{x'\sim\mathbb{P}_h(\cdot|\texttt{Unf}(\Upsilon_h))}[|f_h^\star(\upsilon_i, x') - f_h^\star(\upsilon_j, x')|]$. For any $\upsilon_i, \upsilon_h \in \Upsilon_h$ it holds that

$$\Delta^\star(i,j) = \begin{cases} \frac{2}{|\Upsilon_h|} & \upsilon_i \textit{ and } \upsilon_j \textit{ lead to different endogenous state} \\ 0 & o.w. \end{cases} \tag{5}$$

Observe that we can learn $f_h^\star$ to sufficiently good accuracy with respect to the distribution $x' \sim \mathbb{P}_h(\cdot|\texttt{Unf}(\Upsilon_h))$ via standard regression guarantees of the MLE (see Theorem 4). Thus, for any $\upsilon_i, \upsilon_j \in \Upsilon_h$ we can estimate $\Delta^\star(i,j)$ and deduce whether they lead to the same endogenous state, when $\Delta^\star(i,j)$ is small, or not, when $\Delta^\star(i,j)$ is large. This step is performed in the elimination step if PPE, line 8. Thus, we can safely eliminate open-loop policies form Υ_h that reach that same endogenous state and be left with a minimal policy cover Ψ_h for the next time step.

Analysis overview, near deterministic dynamics. The analysis of the deterministic setting can be generalized to the near deterministic endogenous dynamics by a delicate modification of the above argument.

For this setting, the simplest way we found to extend the argument for the deterministic case goes as follows. We prove via induction that for any $h \in [H]$ the set of open-loop policies Ψ_h is a minimal policy cover of the η *close deterministic endogenous MDP*. Specifically, we use similar arguments as for the deterministic case, while replacing equation 5 with a proper generalization supplied in Lemma 5. Observe that open loop policies are always endogenous policies ; such a policy does not depend on the state and is picked before interacting with the environment. This fact, allows us to use an inverse dynamics objective which filters the exogenous information.

Tho complete the proof, we show that a minimal policy cover of the η close deterministic MDP is an ηH approximate policy cover (see Definition 2). Furthermore, we also show that with this set of policies we can apply the PSDP Bagnell et al. (2004) algorithm to get a near optimal policy when the reward function is an arbitrary function of observations, that might depend on the exogenous part of the state space.

C.3 Proof of Theorem 1

We start by formally defining a minimal policy cover for deterministic dynamics. This definition can be naturally generalized to general MDPs. Nevertheless, since we only study near deterministic dynamics we will only use the next, more specific definition.

Definition 7 (Minimal Policy Cover for Deterministic Endogenous MDP)**.** *Assume that the endogenous dynamics is deterministic. We say that a policy cover Ψ_h is minimal for time step $h \in [H]$ if for every $s \in \mathcal{S}_h$ there exists a unique path $\upsilon \in \Psi_h$ that reaches it, that is $\mathbb{P}(s|\upsilon) = 1$.*

In this section we supply the proof of Theorem 1.

We now prove Theorem 1 by establishing a more general result that will also be helpful when considering planning algorithms (see Appendix D). See that Theorem 1 is a direct consequence of the second statement of the next result result.

Theorem 2 (Sample Complexity: Policy Cover with PPE)**.** *Assume that there exists an $\eta \le \frac{1}{4SH}$ close deterministic MDP for the endogenous dynamics. Let $\delta \in (0,1)$ and assume PPE has access to $O\left(S^2A^2\log\left(\frac{|\mathcal{F}|SAH}{\delta}\right)\right)$ sample for each iteration $h \in [H]$. Then, with probability greater than $1-\delta$ the following holds.*

1. *For any $h \in [H]$ the policy cover Ψ_h is a minimal policy cover of the η close near deterministic MDP of the endogenous dynamics.*
2. *For any $h \in [H]$ the policy cover Ψ_h is ηH approximate policy cover.*

Proof. **First statement.** To prove this result we prove the following inductive argument. Denote by $\mathcal{M}_D$ the η close deterministic MDP, by T_D its transition function, and let $\mathcal{S}_h^D$ be the set of reachable states on $\mathcal{M}_D$. Let G_h be the good event in which at the end of the h^{th} time step for any $s \in \mathcal{S}_h^D$ there exists a unique path $\upsilon \in \Psi_h$ that reaches s on the η close deterministic MDP. More formally, let $\mathbb{P}_h(\cdot|\nu, T_D)$ denote the state distribution of time step h on the η close deterministic MDP when the open loop policy ν is applied. Then, the good event G_h is defined as follows.

$$G_h = \left\{\textit{at the end of the } h^{th} \textit{ time step: } \forall s \in \mathcal{S}_h^D, \exists \textit{ a unique } \nu \in \Psi_h \textit{ s.t. } \mathbb{P}_h(s|\nu, T_D) = 1\right\} \quad (6)$$

Conditioning on $G_{0:H}$ it holds that, for all $h \in [H]$, Ψ_h is a minimal policy cover of the η close deterministic MDP of the endognous dynamics. Hence, to conclude the proof of the first statement, we can prove that $\mathbb{P}(G_{0:H}) \geq 1 - \delta$. To do so, and since $\mathbb{P}(G_0) = 1$, it is sufficient to prove that

$$\mathbb{P}(G_{h+1}|G_{0:h}) \geq 1 - \frac{\delta}{H} \quad (7)$$

Then by an inductive argument it can be proved that $\mathbb{P}(G_{0:h}) \geq 1 - \frac{(h-1)\delta}{H}$. Indeed, the base case holds since $\mathbb{P}(G_0) = 1$, and the induction step holds since

$$\begin{aligned}
&\mathbb{P}(G_1 \cap \cdots \cap G_{h+1}) = \mathbb{P}(G_1 \cap \cdots \cap G_h)\mathbb{P}(G_{h+1}|G_1 \cap \cdots \cap G_h) \\
&\geq \left(1 - \frac{(h-1)\delta}{H}\right)\mathbb{P}(G_{h+1}|G_1 \cap \cdots \cap G_h) && \text{(inductive assumption)} \\
&\geq \left(1 - \frac{(h-1)\delta}{H}\right)\left(1 - \frac{\delta}{H}\right) && \text{(Assuming equation 7)} \\
&\geq 1 - \frac{h\delta}{H}.
\end{aligned}$$

In Lemma 4 we prove that equation 7 holds and establish the theorem.

Second statement. The endogenous minimal policy cover of the η close deterministic MDP of the endogenous dynamics implies an ηH approximate policy cover for the true EX-BMDP. We formally prove the claim in Lemma 3. ☐

Lemma 3 (Translating Policy Covers)**.** *Let $\mathcal{M} = (\mathcal{Z}, \mathcal{A}, T, T_\xi, H, q)$ be an EX-BMDP with η near deterministic endogenous dynamics T_D. Assume that Ψ_h is an endogenous minimal policy cover of the η near deterministic endogenous for the h^{th} time step. Assume that $\mathcal{M}$ and $\mathcal{M}_D$ are η close. Then, Ψ_h is an $h\eta$ approximate policy cover for $\mathcal{M}$.*

Proof. By Lemma 8 it holds for any $s \in \mathcal{S}_h$ that

$$\mathbb{P}_h(s|\pi) \geq \mathbb{P}_h(s|\pi, T_D) - \eta h, \textit{ and } \mathbb{P}_h(s|\pi) \leq \mathbb{P}_h(s|\pi, T_D) + \eta h. \quad (8)$$

If $s \in \mathcal{S}_h^D$ where $\mathcal{S}_h^D$ is the set of reachable states at h time step on T_D, there exists a policy $\pi \in \Psi_h$ such that $\mathbb{P}_h(s|\pi, T_D) = 1$. Thus, due to equation 8, for any $s \in \mathcal{S}_h^D$ there exists a policy $\pi \in \Psi_h$ such

$$\mathbb{P}_h(s|\pi) \geq \mathbb{P}_h(s|\pi, T_D) - \eta h = 1 - \eta h \geq \max_{\pi \in \Pi_d} \mathbb{P}_h(s|\pi) - \eta h. \quad (9)$$

Due to this result, since $\pi \in \Psi_h$ is an endogenous policy and by Proposition 3 it holds that for any $\xi \in \Xi$

$$\begin{aligned}
&\mathbb{P}_h(z|\pi) = \mathbb{P}_h(\xi)\mathbb{P}_h(s|\pi) && \text{(Proposition 3 \& } \pi \in \Pi_d) \\
&\geq \max_{\pi \in \Pi_d} \mathbb{P}_h(\xi)\mathbb{P}_h(s|\pi) - \eta h && \text{(By equation 9)} \\
&= \max_{\pi \in \Pi_d} \mathbb{P}_h(z|\pi) - \eta h && \text{(Proposition 3 \& } \pi \in \Pi_d) \\
&= \max_{\pi \in \Pi_{NS}} \mathbb{P}_h(z|\pi) - \eta h. && \text{(Proposition 4)}
\end{aligned}$$

Hence, for any $z = (s, \xi)$ where $s \in \mathcal{S}_h^D$ it holds that there exists $\pi \in \Psi_h$ such that

$$\mathbb{P}_h(z|\pi) \geq \max_{\pi \in \Pi_{NS}} \mathbb{P}_h(z|\pi) - \eta h.$$

Assume that $s \in \mathcal{S}_h/\mathcal{S}_h^D$, that is, it is reachable in h time steps on the true MDP $\mathcal{M}$ but not on the deterministic MDP $\mathcal{M}_D$. That is, for any π it holds that $\mathbb{P}_h(s|\pi, T_D) = 0$. Thus, again by applying equation 8, it holds that for any $\pi \in \Pi_d$

$$\mathbb{P}_h(s|\pi) \leq \mathbb{P}_h(s|\pi, T_D) + \eta h = \eta h.$$

Fix $\xi \in \Xi$. By multiplying both sides by $\mathbb{P}_h(\xi)$ we get that for any such $z = (s, \xi)$ it holds that

$$\begin{aligned}
&\mathbb{P}_h(\xi) \max_{\pi\in\Pi_d} \mathbb{P}_h(s|\pi) = \max_{\pi\in\Pi_d} \mathbb{P}_h(\xi)\mathbb{P}_h(s|\pi) \\
&= \max_{\pi\in\Pi_d} \mathbb{P}_h(z|\pi) && \text{(Proposition 3 \& } \pi \in \Pi_d) \\
&\leq \eta h.
\end{aligned}$$

On the other hand, by Proposition 4 it holds that

$$\max_{\pi\in\Pi_d} \mathbb{P}_h(z|\pi) = \max_{\pi\in\Pi_{NS}} \mathbb{P}_h(z|\pi).$$

Hence, for any $z = (s, \xi)$ where $s \in \mathcal{S}_h/\mathcal{S}_h^D$ it holds that $\max_{\pi\in\Pi_{NS}} \mathbb{P}_h(z|\pi) \leq \eta H$. Hence, excluding policies that try to reach these state does not violate the definition of an ηh approximate policy cover (see Definition 2).

□

Lemma 4. *The* PPE *algorithm at the h^{th} time step satisfies that*

$$\mathbb{P}(G_h|G_{0:h-1}) \geq 1 - \frac{\delta}{H}$$

where the good event is defined in the proof of Theorem 2. That is, conditioning on the success of PPE *at previous time-steps, with probability greater than $1 - \frac{\delta}{H}$,* PPE *recovers the transition model of $\mathcal{M}_D$ between all reachable states at the h^{th} to the $h + 1^{th}$ level.*

Proof. We will prove the following claims conditioning on $G_{0:h-1}$. Combining the two concludes the proof of the lemma.

1. *Compositionality of policy cover.* The extended policy cover Υ_h is a super set of a policy cover of the h^{th} time step of the η close deterministic transition model.
2. *Elimination succeeds.* Consider the i, j iteration of the for loop in line 6 of Algorithm 1. If $\upsilon_i, \upsilon_j \in \Upsilon_h = \Psi_{h-1} \times \mathcal{A}$ reaches the same endogenous state $s \in \mathcal{S}_h$ on the η-close deterministic MDP, then PPE eliminates ν_j from Υ_h with probability greater than $1 - \frac{\delta}{(SA)^2 H}$.

Taking a union bound on all pairs we get that the elimination procedure succeeds. Thus, PPE outputs a perfect policy cover over $\mathcal{S}_h^D$ with probability greater than $1 - \frac{\delta}{H}$. Thus concluding the proof.

We now prove the two claims.

Claim 1. Conditioning on $G_{0:h-1}$ it holds that Ψ_{h-1} is a perfect policy cover for the η close deterministic transition model. By Lemma 6, utilizing the fact that Ψ_{h-1} is a perfect policy cover, it holds that $\Upsilon_h = \Psi_{h-1} \times \mathcal{A}$ is a policy cover for the h^{th} time step for the η close determinstic dynamics. That is, for any $s \in \mathcal{S}_h^D$ there exists *at least* a single path $\upsilon \in \Upsilon_h$ that reaches s on the η close deterministic MDP.

Claim 2. Let $\upsilon_i, \upsilon_j \in \Upsilon_h$ be a fixed and different paths in the extended policy cover at the h^{th} time step, Υ_h. Let the empirical and the expected disagreement w.r.t. $f \in \mathcal{F}$ be defined as follows.

$$\begin{aligned}
\widehat{\Delta}(i, j; f) &= \frac{1}{N} \sum_{i=1}^{N} |f(\upsilon_i|x_{h,n}) - f(\upsilon_j|x_{h,n})| \\
\Delta(i, j; f) &= \mathbb{E}_{x_h \sim \mathbb{P}(\cdot|\mathcal{U}(\Upsilon_h))} \left[|f(\upsilon_i|x_h) - f(\upsilon_j|x_h)| \right].
\end{aligned}$$

Observe that $\widehat{\Delta}(i,j;f)$ is an average of N i.i.d. and bounded in $[0,1]$ random variables. Furthermore, the expectation of the random variables is exactly $\Delta(i,j;f)$. Thus, for any fixed $\upsilon_i, \upsilon_j \in \Upsilon_h$ and $f \in \mathcal{F}$, given $N = O\left(|\Upsilon_h|^2 \log\left(\frac{1}{\delta}\right)\right)$ samples, due to Hoeffding's inequality, it holds that

$$\left|\widehat{\Delta}(i,j;f) - \Delta(i,j;f)\right| \leq \frac{1}{24|\Upsilon_h|}, \tag{10}$$

with probability greater than $1-\delta$. Applying the union bound on all $f \in \mathcal{F}$ we get that equation 10 holds with probability greater than $1 - \frac{\delta}{|\Upsilon_h|^2 H}$ for all $f \in \mathcal{F}$ given

$$N = O\left(|\Upsilon_h|^2 \log\left(\frac{|\mathcal{F}||\Upsilon_h|H}{\delta}\right)\right).$$

Let $\widehat{\Delta}(i,j) \equiv \widehat{\Delta}(i,j;\hat{f}_h), \Delta(i,j) \equiv \Delta(i,j;\hat{f}_h)$ where $\hat{f}_h$ is the solution of the maximum liklihood objective. Since equation 10 holds w.r.t. $f \in \mathcal{F}$, it implies that

$$\left|\widehat{\Delta}(i,j) - \Delta(i,j)\right| \leq \frac{1}{24|\Upsilon_h|}, \tag{11}$$

with probability greater than $1 - \frac{\delta}{|\Upsilon_h|^2 H}$. Furthermore, notice that

$$\begin{aligned}
&|\Delta^\star(i,j) - \Delta(i,j)| \\
&= \left|\mathbb{E}_{x_h \sim \mathbb{P}(\cdot|\mathcal{U}(\Upsilon_h))}\left[|f^\star_h(\upsilon_i|x_h) - f^\star_h(\upsilon_j|x_h)| - \left|\hat{f}_h(\upsilon_i|x_h) - \hat{f}_h(\upsilon_j|x_h)\right|\right]\right| \\
&\leq \mathbb{E}_{x_h \sim \mathbb{P}(\cdot|\mathcal{U}(\Upsilon_h))}\left[\left|\hat{f}_h(\upsilon_i|x_h) - f^\star_h(\upsilon_i|x_h)\right|\right] + \mathbb{E}_{x_h \sim \mathbb{P}(\cdot|\mathcal{U}(\Upsilon_h))}\left[\left|\hat{f}_h(\upsilon_j|x_h) - f^\star_h(\upsilon_j|x_h)\right|\right] \\
&\leq 2\epsilon,
\end{aligned} \tag{12}$$

where the second relation holds since $||a|-|b|| \leq |a-b|$ due to the triangle inequality, and the third relation holds with probability greater than $1 - \frac{\delta}{|\Upsilon_h|^2 H}$ due to Theorem 4 where $2\epsilon = \frac{2}{24|\Upsilon_h|}$ for $N = O\left(|\Upsilon_h|^2 \log\left(\frac{|\mathcal{F}||\Upsilon_h|^2 H}{\delta}\right)\right)$. Observe that Theorem 4 is applicable due to the realizability Assumption 2.

By combining equation 11 and equation 12 it holds that

$$\left|\widehat{\Delta}(i,j) - \Delta^\star(i,j)\right| \leq 2\epsilon + \frac{1}{24|\Upsilon_h|} = \frac{1}{8|\Upsilon_h|}. \tag{13}$$

By taking a union bound on all $\nu_i, \nu_j \in \Upsilon_h$ we get that equation 13 for all $\nu_i, \nu_j \in \Upsilon_h$ with probability greater than $1 - \frac{\delta}{H}$.

Finally, by Lemma 5 it holds that

$$\begin{cases} |\Delta^\star(i,j)| \leq \frac{1}{4|\Upsilon_h|} & \upsilon_i \textit{ and } \upsilon_j \textit{ reach the same endogenous state on } \mathcal{M}_{D,\eta} \\ |\Delta^\star(i,j)| \geq \frac{1}{|\Upsilon_h|} & o.w. \end{cases}$$

if Ψ_{h-1} is a perfect policy cover, which holds conditioning on G_{h-1}. This fact, together with equation 13 implies that with probability greater than $1 - \frac{\delta}{H}$

$$\begin{cases} \left|\widehat{\Delta}(i,j)\right| \leq \frac{3}{8|\Upsilon_h|} & \upsilon_i \textit{ and } \upsilon_j \textit{ reach the same endogenous state on } \mathcal{M}_{D,\eta} \\ \left|\widehat{\Delta}(i,j)\right| \geq \frac{7}{8|\Upsilon_h|} & o.w. \end{cases}$$

Thus, for any part ν_i, ν_j the elimination process succeeds with probability greater than $1 - \frac{\delta}{H}$ since we set the elimination threshold to be $\frac{5}{8|\Upsilon_h|}$ in line 8 of Algorithm 1. That is, with probability greater than $1 - \frac{\delta}{H}$ at the end of the h^{th} episode we are left with a perfect policy cover on the η close deterministic MDP, conditioning on $G_{0:h-1}$. □

The following lemma highlights the existence of margin for MDPs which are η close to a deterministic MDP for $\eta \leq 1/(4|\mathcal{S}|H)$.

Lemma 5 (Existence of Margin for Inverse Path Prediction Near Deterministic Dynamics)**.** *Assume the transition model is η close to deterministic dynamics. Let Ψ_{h-1} be a perfect policy cover of the η close deterministic dynamics and $\Upsilon_h = \Psi_{h-1} \times \mathcal{A}$ its extension. Let $\Delta^\star(i,j) = \mathbb{E}_{x' \sim \mathbb{P}(\cdot|\mathcal{U}(\Upsilon_h))}[|f^\star(\upsilon_i \mid x') - f^\star(\upsilon_j \mid x')|]$ for a fixed $\upsilon_i, \upsilon_j \in \Upsilon_h$. Then,*

$$\begin{cases} \Delta^\star(i,j) \le \frac{\eta h |\mathcal{S}_h|}{|\Upsilon_h|} & \upsilon_i \text{ and } \upsilon_j \text{ reach the same endogenous state on } \mathcal{M}_{D,\eta} \\ \Delta^\star(i,j) \ge 2\frac{1-2\eta h}{|\Upsilon_h|} & o.w. \end{cases}$$

In particular, if $\eta \le 1/(4|\mathcal{S}_h|h)$, then we have

$$\begin{cases} \Delta^\star(i,j) \le \frac{1}{4|\Upsilon_h|} & \upsilon_i \text{ and } \upsilon_j \text{ reach the same endogenous state on } \mathcal{M}_{D,\eta} \\ \Delta^\star(i,j) \ge \frac{1}{|\Upsilon_h|} & o.w. \end{cases}$$

Proof. For a path υ, let $s(\upsilon)$ denote the state $s \in \mathcal{S}_h$ reached by the path υ in the corresponding deterministic MDP. Then Lemma 8, we have that

$$\mathbb{P}(s_h = s(\upsilon)|\upsilon) \ge 1 - \eta h \quad \text{and for any } s \in \mathcal{S}_h \quad \mathbb{P}(s_h = s \ne s(\upsilon)|\upsilon) \le \eta h. \tag{14}$$

Using this, and the form of the Bayes optimal predictor from Lemma 2, we have that for any two paths υ_i and υ_j such that $s(\upsilon_i) = s(\upsilon_j)$:

$$\begin{aligned} \Delta^\star(i,j) &= \mathbb{E}_{x \sim \mathbb{P}(\cdot|\mathcal{U}(\Upsilon_h))}[|f^\star(\upsilon_i \mid x) - f^\star(\upsilon_j \mid x)|] \\ &= \sum_{s \in \mathcal{S}_h} \mathbb{P}(s|\mathcal{U}(\Upsilon_h))\,|f^\star(\upsilon_i \mid s) - f^\star(\upsilon_j \mid s)| \\ &= \sum_{s \in \mathcal{S}_h} \mathbb{P}(s) \frac{|\mathbb{P}(s|\upsilon_i) - \mathbb{P}(s|\upsilon_j)|}{\sum_{\upsilon \in \Upsilon_h} \mathbb{P}(s|\upsilon)} \\ &= \sum_{s \in \mathcal{S}_h} \mathbb{P}(s) \frac{|\mathbb{P}(s|\upsilon_i) - \mathbb{P}(s|\upsilon_j)|}{|\Upsilon_h|\mathbb{P}(s)} \\ &= \frac{\left(|\mathbb{P}(s(\upsilon_i)|\upsilon_i) - \mathbb{P}(s(\upsilon_i)|\upsilon_j)| + \sum_{s \ne s(\upsilon_i)} |\mathbb{P}(s|\upsilon_i) - \mathbb{P}(s|\upsilon_j)|\right)}{|\Upsilon_h|} \\ &\le \frac{((1 - (1 - \eta h)) + (|\mathcal{S}_h| - 1)\eta h)}{|\Upsilon_h|} \\ &= \frac{\eta h |\mathcal{S}_h|}{|\Upsilon_h|}. \end{aligned}$$

Here the first step follows from the form of $f^\star$ in Lemma 2, which only depends on the latent endogenous state. Second step further expands the definition of $f^\star$ and the next step uses the uniform distribution over paths that we roll-in with. The inequality follows from our earlier bound equation 14.

Similarly, if two paths υ_i and υ_j do not lead to the same endogenous state, then we get

$$\begin{aligned} \Delta^\star(i,j) &= \sum_{s \in \mathcal{S}_h} \frac{|\mathbb{P}(s|\upsilon_i) - \mathbb{P}(s|\upsilon_j)|}{|\Upsilon_h|} \\ &\ge \frac{|\mathbb{P}(s(\upsilon_i)|\upsilon_i) - \mathbb{P}(s(\upsilon_i)|\upsilon_j)|}{|\Upsilon_h|} + \frac{|\mathbb{P}(s(\upsilon_j)|\upsilon_i) - \mathbb{P}(s(\upsilon_j)|\upsilon_j)|}{|\Upsilon_h|} \\ &\ge 2\frac{1 - 2\eta h}{|\Upsilon_h|}, \end{aligned}$$

where the first inequality follows since we can drop the non-negative terms corresponding to the states other than $s(\upsilon_i)$ and $s(\upsilon_j)$, while the second bound follows from (14). $\square$

C.4 HELP LEMMAS

Lemma 6 (Policy Cover in Deterministic Environment)**.** *Let Ψ_{h-1} be a policy cover for time step h and let $\Upsilon_h = \Psi_{h-1} \circ \mathcal{A}$. Then, for any $s \in \mathcal{S}_h$ there exists a $\upsilon \in \Upsilon_h$ such that υ reaches $s \in \mathcal{S}_h$ in h time-steps. That is, Υ_h is a super set of the policy cover.*

Proof. If $s \in \mathcal{S}_h$ it means it can be reached in h time-steps. This implies that exists a state $s_h \in \mathcal{S}_h$ and an action a such that $s_h = f(s_h, a)$. Since for any $s_h \in \mathcal{S}_h$ there exists υ that reaches s_h and since we extend Ψ_{h-1} by taking all possible actions the claim follows. $\square$

D PLANNING WITH APPROXIMATE POLICY COVER IN EX-BMDP

Given access to the policy cover PPE outputs, we can efficiently explore in an EX-BMDP as we show in this section. Interestingly, although the policy cover is obtained by exploring the endogenous part of the state space it still allows to effectively explore the full state space. We now address the problem of learning a near optimal policy w.r.t. a *general reward function* via access to the output of PPE. Later, we consider a more specific case, in which the reward is a function of the *endogenous state space*. For the first case, we apply the PSDP algorithm, while for the latter we can apply the more efficient Value Iteration (VI) procedure.

Planning with a general reward function. In Appendix D.1 we consider a general reward function of the observations, that is $r_h(x, a)$. Given the approximate policy cover PPE outputs we can apply the PSDP Bagnell et al. (2004) algorithm, since it only requires access to a sufficiently good policy cover. Intuitively, given a sufficiently good policy cover we can explore the full state space – both the endogenous and exogenous – and learn an optimal policy based on the dynamics-programming procedure of PSDP.

We make the next policy completeness assumption Dann et al. (2018); Misra et al. (2020), required for the success of the PSDP procedure.

Assumption 3 (Policy Completeness). *For any non-stationary policy represented by* Π*,* $\pi = \pi_1 \circ \cdot \circ \pi_H$ *where* $\{\pi_h\}_{h=1}^H \in \Pi$ *for all* $h \in [H]$ *there exists* $\pi \in \Pi$ *for which*

$$\forall x \in \mathcal{X}_h : \ \pi(x) = \arg\max_a Q_h^\pi(x, a).$$

Given access to such a policy class Π, and given the output policy cover of PPE, PSDP has the following guarantees for an EX-BMDP (see Appendix D.1 for a proof).

Theorem 3 (PSDP for EX-BMDP)**.** *Let* $\epsilon \in (0, 1/2)$ *and* $\delta \in (0, 1)$ *and assume that* $\{\Psi_h\}_{h=1}^H$ *is an* ηH *near deterministic policy cover for the endogenous state space for all* $h \in [H]$*, such that* $\eta \le \frac{1}{4SH}$*. Assume that* PSDP *is given* $N = O\left(\frac{S^2 A H^4 \log\left(\frac{|\Pi| H}{\delta}\right)}{\epsilon^2}\right)$ *in total. Then, with probability greater than* $1 - \delta$*,* PSDP *returns a near optimal policy* $\widehat{\pi}$ *such that*

$$V(\pi^\star) - V(\hat{\pi}) \le \epsilon + H^3 \eta,$$

Although the analysis is standard, the final result demonstrates an interesting phenomena: the sample complexity of PSDP depends on the quality and cardinality of the policy cover, and not on the cardinality of the underlying state space. Specifically, the cardinality of the latent state of an EX-BMDP is $Z = S \times |\Xi|$, whereas PSDP learns a near optimal policy with $O(S^2 A H^2 \log\left(\frac{|\Pi|}{\delta}\right)/\epsilon^2)$ samples. At a higher level, the latter result can be thought of as an extension of Proposition 4, in which, access to an exact policy cover was assumed.

Planning with an endogenous reward function. By further inspecting PPE it can be seen PPE can also return the model of the η close deterministic endogenous dynamics T_D. Instead of merely eliminating paths in PPE, line 8, we can obtain the deterministic model of the endogenous dynamics by tracking which (s_{h-1}, a) pairs reached to s_h at each time step. In case the reward function depends only on the endogenous dynamics, for all $x \in \mathcal{X}$ $r(x, a) = r(\phi^\star(x), a)$, we can simply find the optimal policy via VI w.r.t. the η close deterministic endogenous dynamics.

The performance guarantee of the VI procedure as well as full description of the algorithm is supplied in Appendix D.2.

Proposition 6 (Value Iteration for EX-BMDP)**.** *Let* $\epsilon, \delta \in (0, 1)$ *and assume that* T_D *is the* η *close deterministic model* PPE *outputs. Assume that Algorithm 4 have access to* $N = O\left(\frac{SAH^2 \log\left(\frac{SAH}{\delta}\right)}{\epsilon^2}\right)$ *samples in total. Then, the policy Algorithm 4 outputs is* $6\eta H^3 + \epsilon$ *optimal, that is*

$$V(\pi^\star) - V(\hat{\pi}) \le \epsilon + 6H^2 \eta.$$

Utilizing VI as oppose to PSDP can dramatically reduce the computational burden. Furthermore, see that by Proposition 5, there exists an optimal policy which is endogenous.

D.1 General Observation Based Reward: Policy Search by Dynamic Programming

Algorithm 3 $\texttt{PSDP}(\epsilon, \delta, \{\Psi_t\}_{h=1}^H)$

require: ηH near deterministic policy cover $\{\Psi_h\}_{h=1}^H$, $\epsilon, \delta > 0$ accuracy and confidence level
initialize: $\widehat{\pi} = \{\emptyset\}$
for $h = H, H-1, \cdots, 1$ **do**
  $\mathcal{D} = \emptyset$
  Set $N = O\left(\frac{|\Psi_h| A \log\left(\frac{|\Pi| A}{\delta}\right)}{\epsilon^2}\right)$
  for N times **do**
    $(x, a, p, \hat{Q}_{\widehat{\pi}}) \sim \texttt{Unf}(\Psi_h) \circ \texttt{Unf}(\mathcal{A}) \circ \widehat{\pi}$
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{(x, a, p, \hat{Q}_{\widehat{\pi}})\}$
  $\widehat{\pi}_h = \arg\max_\pi \sum_{(x,a,p,\hat{Q}_{\widehat{\pi}}) \in \mathcal{D}} \left(\frac{\widehat{Q}_{\widehat{\pi}}(x_h, a) \mathbf{1}\{a' = a\}}{1/A}\right)$
  $\widehat{\pi} \leftarrow \widehat{\pi}_h \circ \widehat{\pi}$
return $\widehat{\pi}$

Let $\widehat{\pi}$ be the policy at the beginning of the h^{th} iteration. Hence, it is an $H - (h+1)$ non-stationary policy, defined on steps $h+1, .., H$. From contextual bandit guarantees we have that

$$\mathbb{E}_{x \sim P_h(\cdot | \texttt{Unf}(\Upsilon_h))} \left[Q_h^{\widehat{\pi}}(x; \pi \circ \widehat{\pi}) - Q_h^{\widehat{\pi}}(x; \widehat{\pi}_h \circ \widehat{\pi})\right] \leq \epsilon_{cb} := 4\sqrt{\frac{A}{N} \ln\left(\frac{2|\Pi|}{\delta}\right)}, \tag{15}$$

with probability at least $1-\delta$. The policy $\hat{\pi}_h$ is an output of an offline contextual bandits oracle Langford and Zhang (2008); Agarwal et al. (2014)

$$\hat{\pi}_h \in \arg\max_{\pi \in \Pi} \sum_{x_i, a_i, r_i} \mathbb{E}_{a' \sim \pi(\cdot|x)} \left[\frac{r_i \, \mathbb{1}\{a' = a\}}{1/A}\right],$$

and r_i is a role in policy which takes an action a_i and follows $\hat{\pi}$ until the last time step. In Lemma 9 we formally state the result which is also establishing in (Misra et al., 2020), Proposition 6.

Theorem 3 ($\texttt{PSDP}$ for EX-BMDP)**.** *Let $\epsilon \in (0, 1/2)$ and $\delta \in (0,1)$ and assume that $\{\Psi_h\}_{h=1}^H$ is an ηH near deterministic policy cover for the endogenous state space for all $h \in [H]$, such that $\eta \leq \frac{1}{4SH}$. Assume that $\texttt{PSDP}$ is given $N = O\left(\frac{S^2 A H^4 \log\left(\frac{|\Pi| H}{\delta}\right)}{\epsilon^2}\right)$ in total. Then, with probability greater than $1-\delta$, $\texttt{PSDP}$ returns a near optimal policy $\widehat{\pi}$ such that*

$$V(\pi^\star) - V(\hat{\pi}) \leq \epsilon + H^3 \eta,$$

Proof. From the difference lemma we have:

$$V(\pi^\star) - V(\hat{\pi}) = \sum_{h=1}^H \mathbb{E}_{x \sim P_h(\cdot | \pi^\star)} \left[Q_{h+1}^{\hat{\pi}_{h+1:H}}(x, \pi_h^\star(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x; \widehat{\pi}_h(x))\right]. \tag{16}$$

Next, we fix an $h \in [H]$ and bound each term of the above sum. Remember that, conditioning on the good event, Ψ_h is a minimal policy cover of the η near deterministic endogenous MDP $\mathcal{M}_D$. Let the set of reachable endogenous states, after h time steps, of $\mathcal{M}_D$ be denoted by $\mathcal{S}_h^D$. Then, the following holds.

$$
\begin{aligned}
&\mathbb{E}_{x\sim P_h(\cdot|\pi^\star)}\left[Q_{h+1}^{\hat{\pi}_{h+1:H}}(x,\pi_h^\star(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x;\widehat{\pi}_h(x))\right]\\
&=\sum_{s\in\mathcal{S}_h^D,\xi\in\Xi_h}\mathbb{P}_h(s,\xi|\pi^\star)\mathbb{E}_{x\sim q(\cdot|s,\xi)}\left[Q_{h+1}^{\hat{\pi}_{h+1:H}}(x,\pi_h^\star(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x;\widehat{\pi}_h(x))\right]\\
&\quad+\sum_{s\in\mathcal{S}_h/\mathcal{S}_h^D,\xi\in\Xi_h}\mathbb{P}_h(s,\xi|\pi^\star)\mathbb{E}_{x\sim q(\cdot|s,\xi)}\left[Q_{h+1}^{\hat{\pi}_{h+1:H}}(x,\pi_h^\star(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x;\widehat{\pi}_h(x))\right]\\
&\le\sum_{s\in\mathcal{S}_h^D,\xi\in\Xi_h}\mathbb{P}_h(s,\xi|\pi^\star)\mathbb{E}_{x\sim q(\cdot|s,\xi)}\left[Q_{h+1}^{\hat{\pi}_{h+1:H}}(x,\pi_h^\star(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x;\widehat{\pi}_h(x))\right]\\
&\quad+H\sum_{s\in\mathcal{S}_h/\mathcal{S}_h^D,\xi\in\Xi_h}\mathbb{P}_h(s,\xi|\pi^\star) \qquad\qquad \text{(Values are in } [0,H])\\
&=\sum_{s\in\mathcal{S}_h^D,\xi\in\Xi_h}\mathbb{P}_h(s,\xi|\pi^\star)\mathbb{E}_{x\sim q(\cdot|s,\xi)}\left[Q_{h+1}^{\hat{\pi}_{h+1:H}}(x,\pi_h^\star(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x;\widehat{\pi}_h(x))\right] + H\sum_{s\in\mathcal{S}_h/\mathcal{S}_h^D}\mathbb{P}_h(s|\pi^\star)
\end{aligned}
\tag{17}
$$

where the last relation holds by marginalizing over ξ.

Bound on the first term of equation 17. By a standard PSDP analysis as follows.

$$
\begin{aligned}
&\sum_{s\in\mathcal{S}_h^D,\xi\in\Xi_h}\mathbb{P}_h(s,\xi|\pi^\star)\mathbb{E}_{x\sim q(\cdot|s,\xi)}\left[Q_{h+1}^{\hat{\pi}_{h+1:H}}(x,\pi_h^\star(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x;\widehat{\pi}_h(x))\right]\\
&=\sum_{s\in\mathcal{S}_h^D,\xi\in\Xi_h}\frac{\mathbb{P}_h(s,\xi|\pi^\star)}{\mathbb{P}_h(s,\xi|\mathtt{Unf}(\Psi_h))}\mathbb{P}_h(s,\xi|\mathtt{Unf}(\Psi_h))\mathbb{E}_{x\sim q(\cdot|s,\xi)}\left[Q_{h+1}^{\hat{\pi}_{h+1:H}}(x,\pi_h^\star(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x;\widehat{\pi}_h(x))\right]\\
&\le\sum_{s\in\mathcal{S}_h^D,\xi\in\Xi_h}\frac{\mathbb{P}_h(s,\xi|\pi^\star)}{\mathbb{P}_h(s,\xi|\mathtt{Unf}(\Psi_h))}\mathbb{P}_h(s,\xi|\mathtt{Unf}(\Psi_h))\mathbb{E}_{x\sim q(\cdot|s,\xi)}\left[\max_{\pi(x)}Q_{h+1}^{\hat{\pi}_{h+1:H}}(x,\pi(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x;\widehat{\pi}_h(x))\right]\\
&\le||\frac{\mathbb{P}_h(\cdot|\pi^\star)}{\mathbb{P}_h(\cdot|\mathtt{Unf}(\Psi_h))}||_\infty\sum_{s\in\mathcal{S}_h^D,\xi\in\Xi_h}\mathbb{P}_h(s,\xi|\mathtt{Unf}(\Psi_h))\mathbb{E}_{x\sim q(\cdot|s,\xi)}\left[\max_{\pi(x)}Q_{h+1}^{\hat{\pi}_{h+1:H}}(x,\pi(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x;\widehat{\pi}_h(x))\right]\\
&\le||\frac{\mathbb{P}_h(\cdot|\pi^\star)}{\mathbb{P}_h(\cdot|\mathtt{Unf}(\Psi_h))}||_\infty\mathbb{E}_{x\sim\mathbb{P}_h(\cdot|\mathtt{Unf}(\Psi_h))}\left[\max_{\pi(x)}Q_{h+1}^{\hat{\pi}_{h+1:H}}(x,\pi(x)) - Q_{h+1}^{\hat{\pi}_{h+1:H}}(x;\widehat{\pi}_h(x))\right]\\
&\le||\frac{\mathbb{P}_h(\cdot|\pi^\star)}{\mathbb{P}_h(\cdot|\mathtt{Unf}(\Psi_h))}||_\infty\epsilon_{cb}.
\end{aligned}
\tag{18}
$$

The forth relation holds as we added additional positive terms, and $||\frac{\mathbb{P}_h(\cdot|\pi^\star)}{\mathbb{P}_h(\cdot|\mathtt{Unf}(\Psi_h))}||_\infty\ge 0$. The fifth relation holds by the contextual bandits guarantee equation 15 and due to the policy completeness Assumption 3 by which the maximizer is contained in Π.

We now bound the importance sampling ration.

$$
||\frac{\mathbb{P}_h(\cdot|\pi^\star)}{\mathbb{P}_h(\cdot|\mathtt{Unf}(\Psi_h))}||_\infty=\max_{s\in\mathcal{S}_h^D,\xi\in\xi}\frac{\mathbb{P}_h(s,\xi|\pi^\star)}{\mathbb{P}_h(s,\xi|\mathtt{Unf}(\Psi_h))}.
$$

First, by Bayes' theorem it holds that

$$
\mathbb{P}_h(s,\xi|\pi^\star)=\mathbb{P}_h(s|\pi^\star,\xi)\mathbb{P}_h(\xi|\pi^\star)=\mathbb{P}_h(s|\pi^\star,\xi)\mathbb{P}_h(\xi),
$$

since the exogenous state is not affected by the policy.

Second, observe that

$$
\mathbb{P}_h(s,\xi|\mathtt{Unf}(\Psi_h))=\mathbb{P}_h(\xi)\mathbb{P}_h(s|\mathtt{Unf}(\Psi_h))
$$

since the PPE outputs an endogenous policy cover (thus, we can apply Proposition 3). Furthermore, for any endogenous state $s\in\mathcal{S}_h^D$ it holds that

$$
\mathbb{P}_h(s|\mathtt{Unf}(\Psi_h))=\frac{1}{|\Psi_h|}\sum_{\nu\in\Psi_h}\mathbb{P}_h(s|\nu)\ge\frac{1-\eta h}{\Psi_h},
\tag{19}
$$

since there exists $\nu \in \Psi_h$ such that

$$\mathbb{P}_h(s|\nu) \geq \mathbb{P}_h(s|\nu, T_D) - \eta h = 1 - \eta h.$$

Remember that Ψ_h is a (minimal) policy cover of the near deterministic dynamics $\mathcal{M}_D$ and $s \in \mathcal{S}_h^D$ which means there exists $\upsilon \in \Psi_h$ such that $\mathbb{P}_h(s|\nu, T_D) = 1$. Thus, we get

$$\begin{aligned}
||\frac{\mathbb{P}_h(\cdot|\pi^\star)}{\mathbb{P}_h(\cdot|\texttt{Unf}(\Psi_h))}||_\infty &= \max_{s\in\mathcal{S}_h^D,\xi\in\xi} \frac{\mathbb{P}_h(s|\pi^\star,\xi)}{\mathbb{P}_h(s|\texttt{Unf}(\Psi_h))} \\
&\leq \max_{s\in\mathcal{S}_h^D,\xi\in\xi} \frac{\mathbb{P}_h(s|\pi^\star,\xi)}{\mathbb{P}_h(s|\texttt{Unf}(\Psi_h))} \\
&\leq \frac{|\Psi_h|}{1-\eta h}. && \text{(By equation 19)}
\end{aligned}$$

Bound on the second term of equation 17. Observe that for any $s \in \mathcal{S}_h/\mathcal{S}_h^D$ it holds that

$$\mathbb{P}_h(s|\pi, P_{D,\eta}) = 0, \tag{20}$$

since s is a non-reachable state in the η close deterministic MDP $\mathcal{M}_D$. Using this, and utilizing Lemma 8 we get,

$$\begin{aligned}
&\sum_{s\in\mathcal{S}_h/\mathcal{S}_h^D} \mathbb{P}_h(s|\pi^\star) \\
&= \sum_{s\in\mathcal{S}_h/\mathcal{S}_h^{D,\eta}} |\mathbb{P}_h(s|\pi^\star)| = \sum_{s\in\mathcal{S}_h/\mathcal{S}_h^{D,\eta}} |\mathbb{P}_h(s|\pi^\star) - \mathbb{P}_h(s|\pi^\star, T_D)| && \text{(By equation 20)} \\
&\leq \sum_{s\in\mathcal{S}_h} |\mathbb{P}_h(s|\pi^\star) - \mathbb{P}_h(s|\pi, T_D)| \\
&= ||\mathbb{P}_h(\cdot|\pi^\star) - \mathbb{P}_h(\cdot|\pi^\star, T_D)|| \\
&\leq h\eta. && \text{(By Lemma 8, holds for any } \pi)
\end{aligned}$$

Thus, the second term of equation 17 is bounded by

$$H \sum_{s\in\mathcal{S}_h/\mathcal{S}_h^D} \mathbb{P}_h(s|\pi^\star) \leq H^2\eta.$$

Combining the bounds on the first and second term of equation 17. Plugging these bounds and using $|\Psi_h| \leq S$ (condiniong on the good event of PPE), we conclude that

$$V(\pi^\star) - V(\hat{\pi}) \leq \frac{SH}{1-\eta h}\epsilon_{cb} + H^3\eta \leq 2SH\epsilon_{cb} + H^3\eta,$$

since $\eta \leq \frac{1}{4SH} \leq \frac{1}{4H}$ by assumption and since $S \geq 1$. Choosing

$$N = O\left(\frac{S^2H^4\log\left(\frac{|\Pi|H}{\delta}\right)}{\epsilon^2}\right),$$

and applying a union bound such that Lemma 9 holds for any $h \in [H]$ we conclude the proof. □

D.2 Endogenous State Reward: Value Iteration

In this section, we suggest a computationally easier procedure to get a near optimal policy given the output of PPE. If the reward function depends only on the endogenous state then there exists an endogenous optimal policy (see Proposition 5). We show we can utilize this fact and get a near optimal policy by applying the value iteration procedure (VI).

First, we claim the following: PPE *can also return the* η *near deterministic MDP of the endogenous dynamics* $\mathcal{M}_D$. Consider the elimination step of line 8. When such elimination occurs, it implies that two paths υ_i, υ_j lead to the same endogenous state. Conditioning on the good event, there is a

Algorithm 4 $\texttt{VI}(\epsilon, \delta, T_D, \Psi_h)$

require: $\epsilon, \delta > 0$ $\{\Psi_h\}_{h=1}^H$, T_D model of the near deterministic endogenous MDP,
initialize: Allocate $\{Q_h(s,a)\}_{h\in[H], s\in[|\Psi_h|], a\in[A]}$
for $h = H, H-1, ..., 1$ **do**
  Set $N = O\left(\frac{\log\left(\frac{|\Psi_h|A}{\delta}\right)}{\epsilon^2}\right)$
  for $s \in [\Psi_h], a \in [a]$ **do**
    Collect a dataset $\mathcal{D}_{s,a}$ of N *i.i.d.* reward instances by following $\nu_s \circ a$ where $\nu = idx(s)$
    Estimate immediate reward $\hat{r}_h(s,a) = \frac{1}{N}\sum_{R_n \in \mathcal{D}_{s,a}} R_n$
return Optimal policy w.r.t. $\{\hat{r}_h(s,a)\}_{h\in[H], s_h\in|\Psi_h|, a\in\mathcal{A}}$ and T_D via Value Iteration

one-to-one correspondence between $\nu \in \Upsilon_h = \Psi_{h-1} \circ \mathcal{A}$ and (s,a) where $s \in \mathcal{S}_{h-1}$ and endogenous state of the previous time step. This allows us to decode the states from the paths after the elimination step.

Let $s(\upsilon_i)$ and $s(\upsilon_j)$ be the unique states of $\mathcal{M}_D$ that are reached by $\upsilon_{i,h}$ and $\upsilon_{j,h}$ at time step h. Denote also $\upsilon_{i,h} = \upsilon_{i,h-1} \circ a_{i,h}$ and $\upsilon_{j,h} = \upsilon_{j,h-1} \circ a_{j,h}$ where $\upsilon_{l,h-1}$ denotes the first $h-1$ actions of the path $\upsilon_{l,h}$, and $a_{l,h}$ denotes the action at time step h. If $s_{h,ij} = s_h(\upsilon_{i,h}) = s_h(\upsilon_{j,h})$, i.e., $\upsilon_{i,h}$ and $\upsilon_{j,h}$ leads to the same endogenous state, PPE will eliminate one of the paths. In this case, we represent the state reached by both υ_i and υ_j using a single index. Furthermore, by an induction argument, $\upsilon_{i,h-1}$ and $\upsilon_{j,h-1}$ leads to different endogenous states $s(\upsilon_{i,h-1})$ and $s(\upsilon_{j,h-1})$. Thus, we record there is a transition from $(s(\upsilon_{i,h-1}), a_{i,h})$ and $(s(\upsilon_{j,h-1}), a_{j,h})$ to $s_{h,ij}$.

Thus, instead of merely eliminating paths to create a minimal policy cover at each time step, we can also recover the η-close deterministic dynamics $\mathcal{M}_D$. Given this approximate MDP and an endogenous reward function, we can simply apply VI and recover a near optimal policy. Observe that we recover only a near optimal policy since the true dynamics is not the deterministic dynamics of $\mathcal{M}_D$. However, the suboptimality gap can be easily bounded via a value difference lemma, since the models are η close.

Proposition 6 (Value Iteration for EX-BMDP). *Let $\epsilon, \delta \in (0,1)$ and assume that T_D is the η close deterministic model* PPE *outputs. Assume that Algorithm 4 have access to $N = O\left(\frac{SAH^2\log\left(\frac{SAH}{\delta}\right)}{\epsilon^2}\right)$ samples in total. Then, the policy Algorithm 4 outputs is $6\eta H^3 + \epsilon$ optimal, that is*

$$V(\pi^\star) - V(\hat{\pi}) \le \epsilon + 6H^2\eta.$$

Proof. **First step.** The reward is approximated well. We saw that PPE returns the η near deterministic model $\mathcal{M}_D$ (with probability greater than $1-\delta$) denoted by T_D. Fix $h \in [H]$. To estimate the endogenous reward $(\hat{s}, a)$ up to accuracy $\epsilon > 0$ it is sufficient to follow every $\nu \in \Psi_h$ and apply action $a \in \mathcal{A}$, where ν is the policy that reaches a unique state $s \in \mathcal{S}_h^D$. This holds, since there is a one-to-one correspondence between states recovered by PPE and the reachable state space, in h time steps, of the η close deterministic MDP $\mathcal{M}_D$.

The estimated reward function is given by $r_D(\hat{s}_h, a) = \mathbb{E}[R(\hat{s}_h, a_h) \mid \nu \circ a]$, where $\hat{s}_h$ is equivalent to some open-loop policy $\nu \in \Psi_h$. It holds that,

$$\begin{aligned}
r_D(\hat{s}_h, a) &= \mathbb{P}_h(s_h = \hat{s}_h, a_h = a|\nu\circ a)\mathbb{E}[R(s_h = \hat{s}_h, a = a) \mid \nu\circ a, \hat{s}_h, a_h] + \mathbb{P}_h(s_h \ne \hat{s}_h|\nu\circ a) \\
&= \mathbb{P}_h(s_h = \hat{s}_h, a_h = a|\nu\circ a)\mathbb{E}[R(s_h = \hat{s}_h, a = a) \mid \hat{s}_h, a_h] + \mathbb{P}_h(s_h \ne \hat{s}_h|\nu\circ a) \\
&\le r_h(\hat{s}_h, a) + \eta H. \qquad \text{(Lemma 8)}
\end{aligned}$$

On the other hand, via similar reasoning and since $R \in [0,1]$,

$$\mathbb{E}[R(\hat{s}_h, a_h) \mid \nu\circ a] \ge (1-\eta H) r_h(\hat{s}_h, a) - \eta H \ge r_h(\hat{s}_h, a) - 2H\eta.$$

Thus,

$$|\mathbb{E}[R(\hat{s}_h, a_h) \mid \nu\circ a] - r_h(\hat{s}_h, a)| \le 2\eta H.$$

Since the estimated reward is $\widehat{r}_h(\hat{s}_h, a) = \frac{1}{N}\sum_n R_n$ where $R_n \in [0,1]$ and $\mathbb{E}[R_N] = \mathbb{E}[R(\hat{s}_h, a_h) \mid \nu \circ a]$, we get that given $N = O\left(\frac{H^2 \log(\frac{|\Psi_h|AH}{\delta})}{\epsilon^2}\right)$ samples for each $s \in |\Psi_h|, a \in \mathcal{A}, h \in [H]$,

$$|\widehat{r}_h(\hat{s}_h, a) - r_D(\hat{s}_h, a)| \leq \frac{\epsilon}{2H}$$

by Hoeffding's inequality and applying a union bound. Since $S \geq |\Psi_h|$ for any h the total needed number of samples is

$$N = O\left(\frac{SAH^2 \log(\frac{SAH}{\delta})}{\epsilon^2}\right).$$

Bounding gap from optimality. Let $\mathcal{S}_h^D$ be the set of reachable states on the η close endogenous deterministic MDP $\mathcal{M}_D$. Denote by $\mathcal{M}$ the true endogenous MDP. By the first step, and the fact both models are η close it holds that for any $h \in [H], s \in \mathcal{S}_h^D, a \in \mathcal{A}$

$$|\hat{r}(\hat{s}, a) - r(\hat{s}, a)| \leq \frac{\epsilon}{2H} + 2\eta H$$
$$||T(\cdot|\hat{s}, a) - T_D(\cdot|\hat{s}, a))|| \leq \eta.$$

By the value difference lemma, it holds that for any $s \in \mathcal{S}$ it holds that

$$|V_1^\pi(s; \mathcal{M}) - V_1^\pi(s; \mathcal{M}_D)|$$
$$\leq \mathbb{E}\left[\sum_{h'=h}^{H} |r(s_{h'}, a_{h'}) - \hat{r}(s_{h'}, a_{h'})| + \left|(T - T_D)(\cdot \mid s_{h'}, a_{h'})^\top V_{h'+1}^\pi(\cdot; \mathcal{M})\right| \mid s_h = s, \pi, T_D\right]$$
$$\leq \mathbb{E}\left[\sum_{h'=h}^{H} |r(s_{h'}, a_{h'}) - \hat{r}(s_{h'}, a_{h'})| + ||(T - T_D)(\cdot \mid s_{h'}, a_{h'})||_1 ||V_{h'+1}^\pi(\cdot; \mathcal{M})||_\infty \mid s_h = s, \pi, T_D\right]$$
(Holder's inequality)
$$\leq 3\eta H^2 + \epsilon/2,$$

since for all $s \in \mathcal{S}$ $V_{h'+1}^\pi(s) \in [0, H]$, and the transition model is η close. Thus, an optimal policy on $\mathcal{M}_D$, denoted by $\widehat{\pi}$ is near optimal on the true MDP.

$$V_1^{\hat{\pi}}(s; \mathcal{M}) - V_1^{\pi^\star}(s; \mathcal{M}) \geq V_1^{\hat{\pi}}(s; \mathcal{M}_D) - V_1^{\pi^\star}(s; \mathcal{M}_D) - 6\eta H^3 - \epsilon,$$

since $\hat{\pi}$ is optimal on $\mathcal{M}_D$. This implies that

$$\begin{aligned} &V_1^{\hat{\pi}}(s) \geq V_1^{\pi^\star}(s; \mathcal{M}) - 6\eta H^3 - \epsilon \\ &\to \sum_s \mu(s) V_1^{\hat{\pi}}(s; \mathcal{M}) \geq \sum_s \mu(s) V_1^{\hat{\pi}}(s; \mathcal{M}) - 6\eta H^3 - \epsilon, \end{aligned} \tag{21}$$

since $\sum_s \mu(s) = 1$ and $\mu(s) \geq 0$. Observe that, by the law of total probability, for both $\pi = \pi^*, \hat{\pi}$ it holds that

$$\sum_{s_1} \mu(s_1) V_1^\pi(s_1; \mathcal{M}) = \sum_{h=2}^{H} \sum_{s_h} \sum_{s_1} \mu(s_1) \mathbb{P}_h(s_h|s_1, \pi) r_\pi(s_h) = \sum_{h=2}^{H} \sum_{s_h} \mathbb{P}_h(s_h|\pi) r_\pi(s_h)$$

Furthermore, observe that $\hat{\pi}$ is an open-loop policy and, thus, it is an endogenous policy. Furthermore, by Proposition 5 $\pi^\star$ can be chosen to be an endogenous policy. This implies that for any $h \in [H], z \in \mathcal{Z}$ such that $z = (s, \xi)$

$$\mathbb{P}_h(z|\pi^\star) = \mathbb{P}_h(s|\pi)\mathbb{P}_h(\xi)$$

for $\pi = \pi^*, \hat{\pi}$. By this fact, for both $\pi = \pi^*, \hat{\pi}$, it holds that

$$\begin{aligned} &\sum_{h=2}^{H} \sum_{s_h} \mathbb{P}_h(s_h|\pi) r_\pi(s_h) = \sum_{h=2}^{H} \sum_{s_h} \left(\sum_{\xi_h} \mathbb{P}(\xi_h)\right) \mathbb{P}_h(s_h|\pi) r_\pi(s_h) \\ &= \sum_{h=2}^{H} \sum_{s_h} \sum_{\xi_h} \mathbb{P}_h(\xi) \mathbb{P}_h(s_h|\pi) r_\pi(s_h) \\ &= \sum_{h=2}^{H} \sum_{z_h} \mathbb{P}_h(z_h|\pi) r_\pi(s_h) = V_1(\pi) \end{aligned}$$

Thus, we get that

$$\sum_{s_1} \mu(s_1) V_1^{\pi}(s_1; \mathcal{M}) = V_1(\pi), \tag{22}$$

where $V_1(\pi)$ is the value of π on the joint endogenous-exogenous state space.

Observe that by Proposition 5, when the reward is a function of the endogenous state space, an optimal policy of $\mathcal{M}$ is also an optimal policy on the joint endogenous-exogenous state space. Thus, combining equation 22 and equation 21 implies the result,

$$V(\hat{\pi}) \geq V^{\star} - 6\eta H^2 - \epsilon,$$

where $V^{\star}$ is the optimal value on the join endogenous-exogenous state space. $\square$

E Existing Results

Lemma 7 (E.g. Dann et al. (2017), Lemma E.15)**.** *Consider two MDPs $\mathcal{M} = (\mathcal{S}, \mathcal{A}, T, r, H)$ and $\mathcal{M}' = (\mathcal{S}, \mathcal{A}, T', r', H)$. For any policy π and any s, h the following relation holds:*

$$\begin{aligned} &V_h^\pi(s; \mathcal{M}) - V_h^\pi(s; \mathcal{M}') \\ &= \mathbb{E}\left[\sum_{h'=h}^{H} (r_h(s_{h'}, a_{h'}) - r'_{h'}(s_{h'}, a_{h'})) + (T - T')(\cdot \mid s_{h'}, a_{h'})^\top V_{h'+1}^\pi(\cdot; \mathcal{M}) \mid s_h = s, \pi, T'\right]. \end{aligned}$$

The following result is a consequence of existing bounds (e.g. Rosenberg and Mansour (2019), Lemma 28 or Efroni et al. (2021), Lemma 21). We provide the proof of this result for completeness and show it is a direct consequence of the value difference lemma.

Lemma 8 (Perturbations of State Action Frequency Measure)**.** *Let π be a fixed policy. Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be two MDPs with η close transition model, i.e., for any s, a*

$$||T_1(\cdot|s, a) - T_2(\cdot|s, a)|| \leq \eta.$$

Then, for any $h \in [H]$ it holds that

$$||\mathbb{P}_h(\cdot|\mathcal{M}_1, \pi) - \mathbb{P}_h(\cdot|\mathcal{M}_2, \pi)||_1 \leq \eta h.$$

Proof. Fix an $s \in \mathcal{S}_h$, and denote

$$\mathbb{P}_h(s|\mathcal{M}_1, \pi) = \mathbb{E}[\mathbb{1}\{s_h = s\} |\mathcal{M}_1], \mathbb{P}_h(s|\mathcal{M}_2, \pi) = \mathbb{E}[\mathbb{1}\{s_h = s\} |\mathcal{M}_2].$$

This implies that

$$\begin{aligned} &\mathbb{P}_h(s|\mathcal{M}_1, \pi) - \mathbb{P}_h(s|\mathcal{M}_2, \pi) = \mathbb{E}\left[\mathbb{1}\{s_h = s\} \mid \mathcal{M}_1\right] - \mathbb{E}\left[\mathbb{1}\{s_h = s\} \mid \mathcal{M}_2\right] \\ &= \mathbb{E}\left[\sum_{h'=1}^{h} \sum_{s' \in \mathcal{S}_{h'+1}} \left(T_{1,h'}(s'|s_{h'}, a_{h'}) - T_{2,h'}(s'|s_{h'}, a_{h'})\right) V_{h'+1}(s'; \mathcal{M}_2, s) \mid T_1, \pi\right] \end{aligned} \tag{23}$$

where the last relation holds by the value difference lemma (Lemma 7). See that

$$V_{h'+1}(s'; \mathcal{M}_2, s) = \mathbb{E}[\mathbb{1}\{s_h = s\} |\mathcal{M}_2, s', \pi] \geq 0, \tag{24}$$

which also implies that for any $s' \in \mathcal{S}_{h'+1}$

$$\sum_{s \in \mathcal{S}_h} V_{h'+1}(s'; \mathcal{M}_2, s) = \mathbb{E}[\sum_{s \in \mathcal{S}_h} \mathbb{1}\{s_h = s\} |\mathcal{M}_2, s', \pi] = 1 \tag{25}$$

since the indicator is non-zero only on a single state at the h^{th} time step. By utilizing the above we get

$$\begin{aligned} &||\mathbb{P}_\pi(\cdot|\mathcal{M}_1) - \mathbb{P}_\pi(\cdot|\mathcal{M}_2)||_1 = \sum_{s \in \mathcal{S}_h} |\mathbb{P}_h(s|\mathcal{M}_1, \pi) - \mathbb{P}_h(s|\mathcal{M}_2, \pi)| \\ &= \sum_{s \in \mathcal{S}_h} \left| \mathbb{E}\left[\sum_{h'=1}^{h} \sum_{s' \in \mathcal{S}_{h'+1}} \left(T_{1,h'}(s'|s_{h'}, a_{h'}) - T_{2,h'}(s'|s_{h'}, a_{h'})\right) V_{h'+1}(s'; \mathcal{M}_2, s) | T_1, \pi\right]\right| && \text{(By equation 23)} \\ &\leq \mathbb{E}\left[\sum_{h'=1}^{h} \sum_{s' \in \mathcal{S}_{h'+1}} |T_{1,h'}(s'|s_{h'}, a_{h'}) - T_{2,h'}(s'|s_{h'}, a_{h'})| \sum_{s \in \mathcal{S}_h} |V_{h'+1}(s'; \mathcal{M}_2, s)| \, | T_1, \pi\right] && \text{(Triangle's inequality)} \\ &= \mathbb{E}\left[\sum_{h'=1}^{h} \sum_{s' \in \mathcal{S}_{h'+1}} |T_{1,h'}(s'|s_{h'}, a_{h'}) - T_{2,h'}(s'|s_{h'}, a_{h'})| \sum_{s \in \mathcal{S}_h} V_{h'+1}(s'; \mathcal{M}_2, s) | T_1, \pi\right] && \text{(By equation 24)} \\ &= \mathbb{E}\left[\sum_{h'=1}^{h} \sum_{s' \in \mathcal{S}_{h'+1}} |T_{1,h'}(s'|s_{h'}, a_{h'}) - T_{2,h'}(s'|s_{h'}, a_{h'})| \, | T_1, \pi\right] && \text{(By equation 25)} \\ &\leq \eta h. \end{aligned}$$

where the last relation holds by assumption since by $\mathcal{M}_1$ and $\mathcal{M}_2$ are η close in the L_1 norm,

$$\sum_{s' \in \mathcal{S}_{h'+1}} |T_{1,h'}(s'|s_{h'}, a_{h'}) - T_{2,h'}(s'|s_{h'}, a_{h'})| = ||T_{1,h'}(\cdot|s_{h'}, a_{h'}) - T_{2,h'}(\cdot|s_{h'}, a_{h'})|| \leq \eta.$$

□

Lemma 9 (see Langford and Zhang (2008); Misra et al. (2020))**.** *Let $D = (x_i, a_i, r_i)$ be sampled i.i.d and $r_i \in [0, H]$. Let $\hat{\pi}$ be a solution of the offline contextual bandit optimization routine*

$$\hat{\pi} \in \arg\max_{\pi \in \Pi} \sum_{x_i, a_i, r_i} \mathbb{E}_{a' \sim \pi(\cdot|x)} \left[\frac{r_i \mathbf{1}\left\{a' = a\right\}}{1/A} \right].$$

Then,

$$\mathbb{E}_{x,r}[r(\hat{\pi}(x))] \geq \max_{\pi \in \Pi} \mathbb{E}_{x,r}[r(\pi(x))] - \epsilon_{cb},$$

where $\epsilon_{cb} = 4H\sqrt{\frac{A \log\left(\frac{2|\Pi|}{\delta}\right)}{N}}$ for $\epsilon_{cb} \leq 1/2$.

Theorem 4 (MLE Guarantees,Agarwal et al. (2020a))**.** *Fix $\delta \in (0, 1)$. If $f^\star \in \mathcal{F}$ (realizability assumption), then the maximum likelihood estimator $\hat{f}$ satisfies:*

$$\mathbb{E}_{x' \sim D}\left[\left\|\hat{f}(\cdot \mid x) - f^\star(\cdot \mid x)\right\|_{TV}\right] \leq \epsilon = \sqrt{\frac{2}{N} \ln \frac{|\mathcal{F}|}{\delta}},$$

with probability at least $1 - \delta$.

F EXPERIMENT DETAILS

F.1 ALGORITHM DETAILS.

We describe the implementation details of various algorithms for the combination lock problem. We describe the implementation details of PPE, which provably solves this task, as well as three alternatives, Homer, PPO and PPO + RND. For each algorithm, we do grid search over their most crucial hyperparameters. We measure performance based on number of episodes needed to achieve a mean regret of at most $^{V(\pi^\star)}/_2$. We run each experiment 5 times with different seeds and report median performance. For each value of h, we select the hyperparameter with the best median value.

PPE Details. Our implementation of PPE is almost identical to the pseudocode in Algorithm 1. For each time step h, we collect a dataset of size N and solve a multi-class classification problem (line 3). We train the model with mini-batches and use Adam optimization. We perform gradient clipping to limit the norm of gradient to a given value κ_n. We separate a certain percent (q_{pct}) of the dataset and use it as a validation set. We train the model for a maximum number of epochs (n_{max}) and compute the performance on the validation set after each epoch. We stop when the performance stops improving for κ_p number of epochs where κ_p term is called *patience*. We use the model with the best performance on the validation set. All parameters are initialized randomly and trained by the algorithm.

We use a two-layer feed-forward network to model the function class $\mathcal{F}$. We use Leaky-ReLu non-linearity and use hidden dimension (h_{dim}) of 56. The last layer applies a linear transformation to map the hidden vector to a vector of size equal to the number of paths to predict over. Finally, we apply a softmax layer to convert this vector to a probability distribution.

Hyperparameter values for PPE for the combination lock problem are described in Table 2 on page 37. We do grid search over $N \in \{500, 2000, 5000\}$.

Hyperparameter	Values
Learning rate	0.001
N	$\{500, 2000, 5000\}$
Batch size	256
Grad clip norm (κ_n)	10
Patience (κ_p)	20
Optimization	Adam
Max epochs (n_{max})	50
Hidden dimension (h_{dim})	56
Validation percent (q_{pct})	20%
Parameter Initialization	PyTorch 1.4 Default

Table 2: Hyperparameter values for PPE

Homer Details. Similar to PPE, Homer (Misra et al., 2020) is an iterative algorithm that learns a policy cover incrementally by learning state-abstraction per time step. Homer learns the state-abstraction using noise-contrastive estimation by training a model to predict if a given transition (x, a, x') is causal or acausal. The algorithm creates a dataset of N quads (x, a, x', y) where (x, a, x') is a transition and $y = 1$ implies that it is causal and $y = 0$ implies it is acausal. To describe the data collection process, we define the sampling procedure $(x, a, x') \sim \texttt{Unf}(\Psi_{h-1}) \circ \texttt{Unf}(\mathcal{A})$ which first samples $\pi \sim \texttt{Unf}(\Psi_{h-1})$ and follows π till time step $h-1$ to observe x, and then take an action $a \sim \texttt{Unf}(\mathcal{A})$, and observe $x' \sim T(\cdot \mid x, a)$. Homer generates a single datapoint by sampling two transitions $(x^{(1)}, a^{(1)}, x'^{(1)})$ and $(x^{(1)}, a^{(1)}, x'^{(1)}$ from $\texttt{Unf}(\Psi_{h-1}) \circ \texttt{Unf}(\mathcal{A})$. It then samples $y \sim \texttt{Unf}(\{0, 1\})$ and if $y = 1$ then we add $(x^{(1)}, a^{(1)}, x'^{(1)}, y)$ to the dataset, otherwise, we add $(x^{(1)}, a^{(1)}, x'^{(2)}, y)$. The learning approach uses a budget N_{budget} for the number of abstract states to learn at each time step. We use the code provided to us by authors with their permission. We use the model architecture proposed by Misra et al. (2020) who solve a problem similar to combination lock but without exogenous distractors.

The original Homer algorithm proposes using the general PSDP algorithm to learn the policy cover. However, PSDP is an extremely computationally inefficient algorithm. The authors discussed a more greedy approach, however, that approach still relies on solving a classification problem. Since our domain is deterministic, therefore, we use an even simpler open-loop policy search to learn the policy cover. Formally, given a reward function R and horizon h, we create a dataset of tuples $(\pi \circ a, r)$ by sampling $\pi \in \texttt{Unf}(\Psi_{h-1})$, $a \sim \texttt{Unf}(\mathcal{A})$, and then following the open-loop policy π till time step $h-1$ and then taking action a, and r is the total reward achieved. For each extended open-loop policy $\pi \circ a$, we compute the average total reward $\bar{r}_{\pi\circ a}$ using the collected dataset. Finally, we compute the optimal open-loop policy as $\arg\max_{\pi\circ a} \bar{r}_{\pi\circ a}$. We reuse the dataset from the state abstraction learning for the planning phase.

Hyperparameter values for Homer are provided in Table 3 on page 38. Most computational oracle specific hyperparameters are chosen exactly as for PPE. We do grid search over the value of $n \in \{2000, 5000, 10000\}$. We found that for no values of n, Homer was able to solve the problem for $H = 5$. However, removing exogenous distractors from the problem resulted in success.

Hyperparameter	**Values**
N_{budget}	2
N	$\{2000, 5000, 10000\}$
Learning rate	0.001
Batch size	256
Grad clip norm (κ_n)	10
Patience (κ_p)	20
Optimization	Adam
Max epochs (n_{max})	50
Hidden dimension (h_{dim})	56
Validation percent (q_{pct})	20%
Parameter Initialization	PyTorch 1.4 Default

Table 3: Hyperparameter values for Homer and ID

ID Details. Our implementation of ID baseline is exactly similar to Homer. The only difference is that instead of predicting whether a given transition (x, a, x') is real or imposter, we predict the action a given (x, x') for real transitions. As such, the hyperparameter values for ID is same as in Table 3 on page 38.

PPO and PPO + RND Details. PPO is an actor-critic algorithm which uses entropy-regularization for performing exploration. We also consider augmentation of PPO with exploration bonus using random network distillation (PPO + RND). Random network distillation uses a fixed randomly initialized network g. Given an observation x, the model trains another network to predict the output $g(x)$. An exploration bonus is derived from the prediction error. As the model explores part of the state space, its prediction error for observations emitted by these states goes down, and the model is no longer incentivized to visit these states.

Hyperparameter values for PPO and PPO + RND are shown in Table 4 on page 39. We do grid search over entropy coefficient in $\{0.1, 0.01\}$. For PPO + RND, we also do grid search over RND bonus coefficient in $\{100, 500\}$.

Bisimulation Details. We use the bisimulation code provided by Zhang et al. (2021).[5] We make minimal changes to the repository in order to refactor it for our experiments. Two changes were required, however, to adapt the codebase to our experiments. Firstly, we use a two-layer feed-forward network as encoder and decoder instead of a convolutional neural network (CNN). Since observations in the combination lock experiment consist of a 1D feature vector, therefore, a CNN architecture didn't make much sense. Secondly, the codebase used soft-actor critic for a continuous action space problems. We adapt it to our purpose by discretizing the agent's action. Formally, given a continuous action $u \in [-1, 1]$ generated by the policy, we take the action $a = \lfloor 1/2 \times (u + 1) \times 10 \rfloor$. Since

[5] https://github.com/facebookresearch/deep_bisim4control

Hyperparameter	**Values**
Entropy coefficient	$\{0.1, 0.01\}$
Number of PPO updates	4
Clipping parameter	0.1
Learning rate	0.001
Batch size	256
Grad clip norm (κ_n)	10
Patience (κ_p)	20
Optimization	Adam
Max epochs (n_{max})	50
Hidden dimension (h_{dim})	56
Validation percent (q_{pct})	20%
Parameter Initialization	PyTorch 1.4 Default
only for PPO + RND	
RND bonus coefficient	$\{100, 500\}$

Table 4: Hyperparameter values for PPO and PPO + RND

each discrete action has the same probability of being generated under uniform distribution over u, therefore, this relaxation does not introduce any extra hardness.

Compute Infrastructure and Different Runs. We run experiments on a clusters with a mixture of P40, P100, and V100 GPUs. Each experiment runs on a single GPU in a docker container. We ran each experiment 5 times with different seeds.

F.2 Additional Detail for State Decoding Experiment.

We use $m = 5000$ samples to evaluate the state decoding methods. We run each experiment 10 times with different seed and show mean and standard deviation in Figure 2c. Lastly, we define the distribution $D(x) = \mathbb{E}_{\pi \sim \texttt{Unf}(\Psi)}\left[\mathbb{P}_3(x \mid \pi)\right]$ where Ψ contains an open-loop policy for each of the three states reachable at time step 3, namely, $\{s_{3,a}, s_{3,b}, s_{3,c}\}$.

F.3 Additional Detail for Visual Grid World Experiment.

The agent can take five actions: forward by one step (F), turn-left by 90-degree (L), turn right by 90-degree (R), and two compositional actions LF and RF that first take action R or L, and then take action F. The agent receives a reward of +1 for reaching the goal and a reward of -1 for reaching lava and a small negative reward otherwise.

Our implementation of PPE remains the same as before. However, we use a different architecture for $\mathcal{F}$. We use a two-layer convolutional network with ReLu non-linearity. The first layer applies 16 8×8 kernel with stride 4 and the second layer applies 32 2×2 kernel with stride 2. Finally, we flatten the representation and pass it through a linear layer to a vector of appropriate size. We do not do any image pre-processing and do not use any pre-trained models.

F.4 PPE pseudocode used for experiments

We present the pseudocode that we used in experiments in Algorithm 5. This pseudocode is optimized for problems with deterministic endogenous transition dynamics with reward function that depends on endogenous state. However, as stated earlier, when the reward function depends on exogenous state or observation, then we can use PSDP (Appendix D.1. And when the reward function depends only on endogenous state but the transition dynamics are near-deterministic, then we can use the decoder to learn the transition dynamics and reward and simply use value iteration.

The algorithm is exactly the same as PPE with extra details to show how value iteration is done in practice. In particular, we show how transition function and reward function are estimated. Formally, for extracting the transition function we define a map $\mathcal{U}$ that given an index of a path i contains a set of paths that are *merged* with the i^{th} path, that is υ_i (line 4). Intuitively, two paths merge if they reach

Algorithm 5 $\texttt{PPE}(N)$: Predictive Path Elimination as run in practice

1: Set $\Psi_1 = \{\perp\}$ and $\hat{\mathcal{S}}_1 = \{1\}$ //$\perp$ denotes an empty path
2: **for** $h = 2, \ldots, H+1$ **do**
3: Collect a dataset $\mathcal{D}$ of N *i.i.d.* tuples (x, r, υ) where $\upsilon \sim \texttt{Unf}(\Psi_{h-1} \circ \mathcal{A})$, $r = R(x_{h-1}, a_{h-1})$, and $x \sim \mathbb{P}(x_h \mid \upsilon)$.
4: Define $\mathcal{U}(i) = \{\upsilon_i\}$ for all i in $\{1, 2, \cdots, |\Psi_{h-1} \circ \mathcal{A}|\}$.
5: Solve multi-class classification problem: $\hat{f}_h = \arg\max_{f \in \mathcal{F}} \sum_{(x,r,\upsilon) \in \mathcal{D}} \ln f(\text{idx}(\upsilon) \mid x)$.
6: **for** $1 \le i < j \le |\Psi_{h-1} \circ \mathcal{A}|$ **do**
7: Calculate the path prediction gap: $\widehat{\Delta}(i, j) = \frac{1}{N} \sum_{(x,\upsilon) \in \mathcal{D}} \left| \hat{f}_h(i|x) - \hat{f}_h(j|x) \right|$.
8: **if** $\widehat{\Delta}(i, j) \le \frac{5/8}{|\Upsilon_h|}$ **then,**
9: eliminate path υ with $\text{idx}(\upsilon) = j$. //υ_i and υ_j visit the same state
10: $\mathcal{U}(i) = \mathcal{U}(i) \cup \mathcal{U}(j)$ //paths merged with υ_j are now merged with υ_i
11: Ψ_h is defined as the set of all paths in $\Psi_{h-1} \circ \mathcal{A}$ that have not been eliminated in line 10.
12: Define decoder $\hat{\phi}_h : x \mapsto \min_i \left\{ i \mid \hat{f}_h(i \mid x) \ge \max_j \hat{f}_h(j \mid x) - \mathcal{O}(1/|\Upsilon_h|), i \in |\Upsilon_h| \right\}$.
13: Define $\hat{\mathcal{S}}_h = \{\text{idx}(\nu) \mid \nu \in \Psi_h\}$ //create a state for each path which is not eliminated
14: Define $\hat{T}_{h-1} : \hat{\mathcal{S}}_{h-1} \times \mathcal{A} \to \hat{\mathcal{S}}_h$ as $T(i, a) = j$ if $\upsilon_i^{h-1} \circ a \in \mathcal{U}(j)$ where $\upsilon_i^{h-1} \in \Psi_{h-1}$.
15: Define $\hat{R}_{h-1} : \hat{\mathcal{S}}_{h-1} \times \mathcal{A} \to [0, 1]$ as $\hat{R}_h(i, a) = \texttt{average}\{r \mid (x, r, \upsilon) \in \mathcal{D}, \upsilon = \upsilon_i \circ a\}$
16: Perform value iteration on the tabular MDP $\widehat{\mathcal{M}} = \left(\{\hat{\mathcal{S}}_h\}_{h=1}^{H+1}, \mathcal{A}, \{\hat{T}_h\}_{h=1}^{H}, \{R_h\}_{h=1}^{H}, H, 1 \right)$ and return the optimal open-loop policy $\hat{\pi}$.
17: **return** $\hat{\pi}$, $\widehat{\mathcal{M}}$, $\left\{ \hat{\phi}_h \right\}_{h=2}^{H}$, and $\{\Psi_h\}_{h=2}^{H}$ //return optimal policy, learned latent MDP, decoder, and policy cover

the same endogenous state. Initially, each map is only merged with itself and hence $\mathcal{U}(i)$ is defined as $\{\upsilon_i\}$. When we compare path i and j and eliminate υ_j, we add all the paths that were merged with υ_j as also merged with υ_i (line 10).

We define the recovered latent tabular MDP $\widehat{\mathcal{M}}$ as follows. We firstly, recover the state space, transition dynamics and reward function for each time step separately. We define a state space $\hat{\mathcal{S}}_h$ for the h^{th} time step as containing a state for each path in Ψ_h. We then define a deterministic transition model $\hat{T}_{h-1} : \hat{\mathcal{S}}_{h-1} \times \mathcal{A} \to \hat{\mathcal{S}}_h$ as $\hat{T}(i, a) = j$, if the path $\upsilon_i^{h-1} \circ a \in \Psi_{h-1} \times \mathcal{A}$ with $\upsilon_i^{h-1} = \Psi_{h-1}$, gets merged with path $\upsilon_j \in \Psi_h$ (line 14). Note that we use the superscript $h-1$ to denote that υ_i^{h-1} is not the i^{th} path in $\Psi_{h-1} \times \mathcal{A}$ but in $\Psi h - 2 \times \mathcal{A}$. Further, note that indices of a path are defined with respect to the set $\Psi_t \times \mathcal{A}$ containing them, and remain fixed for the whole training time. Lastly, we compute the reward by using average of reward in the dataset collected corresponding to the appropriate path (line 15).

Finally, we do value iteration on the recovered MDP $\widehat{\mathcal{M}}$ which has a set of states $\hat{\mathcal{S}}_h$ reachable at time step h, a action space $\mathcal{A}$, a transition dynamics $\hat{T}_h$ and reward function $\hat{R}_h$ for the h^{th} time step, a horizon H, and a deterministic start state of $1 \in \hat{\mathcal{S}}_1$ (line 16). As the latent endogenous transition dynamics are deterministic, therefore, an optimal open loop policy exists. We return this optimal policy along with policy cover, decoder, and recovered MDP. These provide pretty much all important objects that we can recover from a given Exogenous Block MDP.