Towards Zero-Shot Generalization in Offline Reinforcement Learning

Zhiyong Wang
The Chinese University of Hong Kong
zhiyongwangwzy@gmail.com

Chen Yang
Department of Computer Science
Indiana University Bloomington
cya2@iu.edu

John C.S. Lui
The Chinese University of Hong Kong
cslui@cse.cuhk.edu.hk

Dongruo Zhou
Department of Computer Science
Indiana University Bloomington
dz13@iu.edu

Abstract

In this work, we study offline reinforcement learning (RL) with zero-shot generalization property (ZSG), where the agent has access to an offline dataset including experiences from different environments, and the goal of the agent is to train a policy over the training environments which performs well on test environments without further interaction. Existing work showed that classical offline RL fails to generalize to new, unseen environments. To address such an issue, we propose new offline RL frameworks with ZSG, based on empirical risk minimization or proximal policy optimization. We prove that our frameworks find the near-optimal policy with ZSG both theoretically and empirically, from general environments to specific settings such as linear Markov decision processes (MDPs). Our result serves as a first step in understanding the foundation of the generalization phenomenon in offline reinforcement learning.

1 Introduction

Offline reinforcement learning (RL) has become increasingly significant in modern RL because it eliminates the need for direct interaction between the agent and the environment; instead, it relies solely on learning from an offline training dataset. However, in practical applications, the offline training dataset often originates from a different environment than the one of interest. This discrepancy necessitates evaluating RL agents in a generalization setting, where the training involves a finite number of environments drawn from a specific distribution, and the testing is conducted on a distinct set of environments from the same or different distribution. This scenario is commonly referred to as the zero-shot generalization (ZSG) challenge which has been studied in online RL[Rajeswaran et al., 2017, Machado et al., 2018, Justesen et al., 2018, Packer et al., 2019, Zhang et al., 2018a,b], as the agent receives no training data from the environments it is tested on.

A number of recent empirical studies [Mediratta et al., 2023, Yang et al., 2023, Mazoure et al., 2022] have recognized this challenge and introduced various offline RL methodologies that are capable of ZSG. Notwithstanding the lack of theoretical backing, these methods are somewhat restrictive; for instance, some are only effective for environments that vary solely in observations[Mazoure et al., 2022], while others are confined to the realm of imitation learning[Yang et al., 2023], thus limiting their applicability to a comprehensive framework of offline RL with ZSG capabilities. Concurrently, theoretical advancements [Bose et al., 2024, Ishfaq et al., 2024] in this domain have explored multi-task offline RL by focusing on representation learning. These approaches endeavor to derive a low-rank representation of states and actions, which inherently requires additional interactions with

Workshop on Aligning Reinforcement Learning Experimentalists and Theorists (ARLET 2024).

the downstream tasks to effectively formulate policies based on these representations. Therefore, we raise a natural question:

Can we design provable offline RL with zero-shot generalization ability?

We propose novel offline RL frameworks that achieve ZSG to address this question affirmatively. Our contributions are listed as follows.

- We propose two meta-algorithms called pessimistic empirical risk minimization (PERM) and pessimistic proximal policy optimization (PPPO) that enable ZSG for offline RL. In detail, both of our algorithms take a pessimistic policy evaluation (PPE) oracle as its component and output policies based on offline datasets from multiple environments. Our result shows that the sub-optimalities of the output policies are bounded by both the supervised learning error, which is controlled by the number of different environments, and the reinforcement learning error, which is controlled by the coverage of the offline dataset to the optimal policy. Please refer to Table 1 for a summary of our results. To the best of our knowledge, our proposed algorithms are the first offline RL methods that provably enjoy the ZSG property.
- Based on the proposed meta-algorithms, we further specify our analysis to a more concrete setting called linear MDPs [Yang and Wang, 2019, Jin et al., 2019]. We show that under the proper coverage assumptions made on the offline dataset distribution, both of our algorithms enjoy the suboptimality gap $O(n^{-1/2}+K^{-1/2}\cdot C_n^*)$, where n is the number of environments in offline dataset, K is the number of trajectories for each environment and C_n^* is a mixed coverage parameter over n environments. [1]. Our result generalizes the convergence result for offline RL in single-environment [Jin et al., 2021].
- We also analyze when existing offline RL approaches like pessimistic-type algorithms [Jin et al., 2021] fail without further algorithm modifications. Such an analysis verifies the necessity of our proposed methods for ZSG property. We also propose numerical results to back up our theoretical claim.

Algorithm	Suboptimality Gap
PERM (our Algo.2)	$\sqrt{\log(\mathcal{N})/n}+n^{-1}\sum_{i=1}^{n}\sum_{h=1}^{H}\mathbb{E}_{i,\pi^*}\left[\Gamma_{i,h}(s_h,a_h)\vert s_1=x_1\right]$
PPPO (our Algo.3)	$\sqrt{\log\vert\mathcal{A}\vert H^2/n}+n^{-1}\sum_{i=1}^{n}\sum_{h=1}^{H}\mathbb{E}_{i,\pi^*}\left[\Gamma_{i,h}(s_h,a_h)\vert s_1=x_1\right]$

Table 1: Summary of our algorithms and their suboptimality gaps, where $\mathcal{A}$ is the action space, H is the length of episode, n is the number of environments in the offline dataset. Note that in the multi-environment setting, π^* is the near-optimal policy w.r.t. expectation (defined in Section 3). $\mathcal{N}$ is the covering number of the policy space Π w.r.t. distance $\mathrm{d}(\pi^1,\pi^2)=\max_{s\in\mathcal{S},h\in[H]}\|\pi_h^1(\cdot|s)-\pi_h^2(\cdot|s)\|_1$. The uncertainty quantifier $\Gamma_{i,h}$ are tailored with the oracle return in the corresponding algorithms (details are in Section 4).

Notation We use lower case letters to denote scalars, and use lower and upper case bold face letters to denote vectors and matrices respectively. We denote by $[n]$ the set $\{1,\ldots,n\}$. For a vector $\mathbf{x}\in\mathbb{R}^d$ and a positive semi-definite matrix $\mathbf{\Sigma}\in\mathbb{R}^{d\times d}$, we denote by $\|\mathbf{x}\|_2$ the vector's Euclidean norm and define $\|\mathbf{x}\|_{\mathbf{\Sigma}}=\sqrt{\mathbf{x}^\top\mathbf{\Sigma}\mathbf{x}}$. For two positive sequences $\{a_n\}$ and $\{b_n\}$ with $n=1,2,\ldots$, we write $a_n=O(b_n)$ if there exists an absolute constant $C>0$ such that $a_n\le Cb_n$ holds for all $n\ge 1$ and write $a_n=\Omega(b_n)$ if there exists an absolute constant $C>0$ such that $a_n\ge Cb_n$ holds for all $n\ge 1$. We use $\widetilde{O}(\cdot)$ to further hide the polylogarithmic factors. We use $(x_i)_{i=1}^n$ to denote sequence $(x_1,...,x_n)$, and we use $\{x_i\}_{i=1}^n$ to denote the set $\{x_1,...,x_n\}$. We use $\mathrm{KL}(p\|q)$ to denote the KL distance between distributions p and q, defined as $\int p\log(p/q)$. We use $\mathbb{E}[x],\mathbb{V}[x]$ to denote expectation and variance of a random variable x.

The remaining parts are organized as follows. In Section 3, we introduce the setting of our work. In Section 4, we introduce our proposed meta-algorithms and provide their theoretical guarantees. Then, in Section 5, we specify our meta-algorithms and analysis to a more concrete linear MDP setting. Finally, in Section 6, we conclude our work and propose some future directions. Due to space constraints, we analyze when existing offline RL approaches [Jin et al., 2021] fail to generalize without further algorithm modifications in Appendix A. We place the experimental results on synthetic and real data in Appendix B, and the analysis of PEVI in Appendix E.2.

[1] Here we only include n, K, C_n^* in our bound for simplicity.

2 Related works

Offline RL Offline reinforcement learning (RL) [Ernst et al., 2005, Riedmiller, 2005, Lange et al., 2012, Levine et al., 2020] addresses the challenge of learning a policy from a pre-collected dataset without direct online interactions with the environment. A central issue in offline RL is the inadequate dataset coverage, stemming from a lack of exploration [Levine et al., 2020, Liu et al., 2020]. A common strategy to address this issue is the application of the pessimism principle, which penalizes the estimated value of under-covered state-action pairs. Numerous studies have integrated pessimism into various single-environment offline RL methodologies. This includes model-based approaches [Rashidinejad et al., 2021, Uehara and Sun, 2021, Jin et al., 2021, Yu et al., 2020, Xie et al., 2021b, Uehara et al., 2021, Yin et al., 2022], model-free techniques [Kumar et al., 2020, Wu et al., 2021, Bai et al., 2022, Ghasemipour et al., 2022, Yan et al., 2023], and policy-based strategies [Rezaeifar et al., 2022, Xie et al., 2021a, Zanette et al., 2021, Nguyen-Tang and Arora, 2024]. To the best of our knowledge, we are the first to theoretically study the generalization ability of offline RL in the contextual MDP setting.

Generalization in online RL There are extensive empirical studies on training online RL agents that can generalize to new transition and reward functions [Rajeswaran et al., 2017, Machado et al., 2018, Justesen et al., 2018, Packer et al., 2019, Zhang et al., 2018a,b, Nichol et al., 2018, Cobbe et al., 2018, Küttler et al., 2020, Bengio et al., 2020, Bertran et al., 2020, Ghosh et al., 2021, Kirk et al., 2023, Juliani et al., 2019, Ajay et al., 2021, Samvelyan et al., 2021, Frans and Isola, 2022, Albrecht et al., 2022, Ehrenberg et al., 2022, Song et al., 2020, Lyle et al., 2022, Ye et al., 2020, Lee et al., 2020, Jiang et al.]. They use techniques including implicit regularizationSong et al. [2020], data augmentation Ye et al. [2020], Lee et al. [2020], uncertainty-driven exploration [Jiang et al.], etc. These works focus on the online RL setting and do not provide theoretical guarantees, thus differing a lot from ours.

There are also some recent works aimed at understanding online RL generalization from a theoretical perspective. Wang et al. [2019] examined a specific class of reparameterizable RL problems and derived generalization bounds using Rademacher complexity and the PAC-Bayes bound. Malik et al. [2021] established lower bounds and introduced efficient algorithms that ensure a near-optimal policy for deterministic MDPs. A more recent work Ye et al. [2023] studied how much pre-training can improve online RL test performance under different generalization settings. To the best of our knowledge, no previous work exists on theoretical understanding of the zero-shot generalization of offline RL.

Our paper is also related to recent works studying multi-task learning in reinforcement learning (RL) [Brunskill and Li, 2013, Tirinzoni et al., 2020, Hu et al., 2021, Zhang and Wang, 2021, Lu et al., 2021, Bose et al., 2024, Ishfaq et al., 2024, Zhang et al., 2023], which focus on transferring the knowledge learned from upstream tasks to downstream ones. Additionally, these works typically assume that all tasks share similar transition dynamics or common representations while we do not. Meanwhile, they typically requires the agent to interact with the downstream tasks, which does not fall into the ZSG regime.

3 Preliminary

Contextual MDP We study *contextual episodic MDPs*, where each MDP $\mathcal{M}_c$ is associated with a context $c \in C$ belongs to the context space C. Furthermore, $\mathcal{M}_c = \{M_{c,h}\}_{h=1}^H$ consists of H different individual MDPs, where each individual MDP $M_{c,h} := (\mathcal{S}, \mathcal{A}, P_{c,h}(s'|s,a), r_{c,h}(s,a))$. Here $\mathcal{S}$ denotes the state space, $\mathcal{A}$ denotes the action space, $P_{c,h}$ denotes the transition function and $r_{c,h}$ denotes the reward function at stage h. We assume the starting state for each $\mathcal{M}_c$ is the same state x_1. In this work, we interchangeablely use "environment" or MDP to denote the MDP $\mathcal{M}_c$ with different contexts.

Policy and value function We denote the policy π_h at stage h as a mapping $(\mathcal{S}\times\mathcal{A}\times[0,1])^{h-1}\times\mathcal{S} \to \Delta(\mathcal{A})$, which maps the history at the h-th stage and the current state to a distribution over the action space. We use $\pi = \{\pi_h\}_{h=1}^H$ to denote their collection. Then for any episodic MDP $\mathcal{M}$, we define the value function for some policy π as

$$V^\pi_{\mathcal{M},h}(x) := \mathbb{E}[r_h + ... + r_H | s_h = x, a_{h'} \sim \pi_{h'}, r_{h'} \sim r_{h'}(s_{h'}, a_{h'}), s_{h'+1} \sim P_{h'}(\cdot|s_{h'}, a_{h'}),\ h' \geq h],$$
$$Q^\pi_{M,h}(x,a) := \mathbb{E}[r_h + ... + r_H | s_h = x, a_h = a, r_h \sim r_h(s_h, a_h), s_{h'} \sim P_{h'-1}(\cdot|s_{h'-1}, a_{h'-1}), a_{h'} \sim \pi_{h'},$$
$$r_{h'} \sim r_{h'}(s_{h'}, a_{h'}),\ h' \geq h+1].$$

For any individual MDP M with reward r and transition dynamic P, we denote its Bellman operator $[\mathbb{B}_M f](x,a)$ as $[\mathbb{B}_M f](s,a) := \mathbb{E}[r_h(s,a) + f(s')|s' \sim P(\cdot|s,a)]$. Then we have the well-known Bellman equation

$$V^\pi_{\mathcal{M},h}(x) = \langle Q^\pi_{\mathcal{M},h}(x,\cdot), \pi_h(\cdot|x)\rangle_{\mathcal{A}},\ Q^\pi_{\mathcal{M},h}(x,a) = [\mathbb{B}_{M_h} V^\pi_{\mathcal{M},h+1}](x,a).$$

For simplicity, we use $V^\pi_{c,h}, Q^\pi_{c,h}, \mathbb{B}_{c,h}$ to denote $V^\pi_{\mathcal{M}_c,h}, Q^\pi_{\mathcal{M}_c,h}, \mathbb{B}_{M_{c,h}}$. We also use $\mathbb{P}_c$ to denote $\mathbb{P}_{\mathcal{M}_c}$, the joint distribution of any potential objects under the $\mathcal{M}_c$ episodic MDP. We would like to find the near-optimal policy π^* w.r.t. expectation, i.e., $\pi^* := \mathrm{argmax}_{\pi\in\Pi} \mathbb{E}_{c\sim C} V^\pi_{c,1}(x_c)$, where Π is the set of collection of Markovian policies, and with a little abuse of notation, we use $\mathbb{E}_{c\sim C}$ to denote the expectation taken w.r.t. the i.i.d. sampling of context c from the context space. Then we would like to characterize the following *zero-shot generalization gap (ZSG gap)*:

$$\text{SubOpt}(\pi) := \mathbb{E}_{c\sim C}\big[V^{\pi^*}_{c,1}(x_1)\big] - \mathbb{E}_{c\sim C}\big[V^{\pi}_{c,1}(x_1)\big].$$

Remark 1 *Compared with ZSG, another line of measurements aim to characterize the generalization performance of RL* with *further interactions with unseen MDPs [Ye et al., 2023]. Clearly, such a setting is stronger than ours, and the additional interactions are often hard to be satisfied in real-world practice. We leave the study of such a setting for future work.*

Offline RL data collection process The data collection process is as follows. An experimenter i.i.d. samples number n of contextual episodic MDP M_i from the context set (*e.g.*, $i \sim C$). For each episodic MDP M_i, the experimenter collects dataset $\mathcal{D}_i := \{(x^\tau_{i,h}, a^\tau_{i,h}, r^\tau_{i,h})^H_{h=1}\}^K_{\tau=1}$ which includes K trajectories. Note that the action $a^\tau_{i,h}$ selected by the experimenter can be arbitrary, and it does not need to follow a specific behavior policy [Jin et al., 2021]. We assume that $\mathcal{D}_i$ is compliant with the episodic MDP $\mathcal{M}_i$, which is defined as follows.

Definition 2 ([Jin et al., 2021]) *For a dataset $\mathcal{D}_i := \{(x^\tau_{i,h}, a^\tau_{i,h}, r^\tau_{i,h})^H_{h=1}\}^K_{\tau=1}$, let $\mathbb{P}_{\mathcal{D}_i}$ be the joint distribution of the data collecting process. We say $\mathcal{D}_i$ is compliant with episodic MDP $\mathcal{M}_i$ if for any $x' \in \mathcal{S}, r', \tau \in [K], h \in [H]$, we have*

$$\begin{aligned}&\mathbb{P}_{\mathcal{D}_i}(r^\tau_{i,h} = r', x^\tau_{i,h+1} = x' | \{(x^j_{i,h}, a^j_{i,h})\}^\tau_{j=1}, \{(r^j_{i,h}, x^j_{i,h+1})\}^{\tau-1}_{j=1})\\ &\quad = \mathbb{P}_i(r_{i,h}(s_h,a_h) = r', s_{h+1} = x' | s_h = x^\tau_h, a_h = a^\tau_h).\end{aligned}$$

In general, we claim $\mathcal{D}_i$ is compliant with $\mathcal{M}_i$ when the conditional distribution of any tuple of reward and next state in $\mathcal{D}_i$ follows the conditional distribution determined by MDP $\mathcal{M}_i$.

4 Provable offline RL with zero-shot generalization

In this section, we propose offline RL with small ZSG gaps. We show that two popular offline RL approaches, *model-based RL* and *policy optimization-based RL*, can output RL agent with ZSG ability, with a pessimism-style modification that encourages the agent to follow the offline dataset pattern.

4.1 Pessimistic policy evaluation

We consider a meta-algorithm to evaluate any policy π given an offline dataset, which serves as a key component in our proposed offline RL with ZSG. To begin with, we consider a general individual MDP and an oracle $\mathbb{O}$, which returns us an empirical Bellman operator and an uncertainty quantifier, defined as follows.

Definition 3 (Empirical Bellman operator and uncertainty quantifier, Jin et al. 2021) *For any individual MDP M, a dataset $\mathcal{D} \subseteq \mathcal{S}\times\mathcal{A}\times\mathcal{S}\times[0,1]$ that is compliant with M, a test function $V_{\mathcal{D}} \subseteq [0,H]^{\mathcal{S}}$ and a confidence level ξ, we have an oracle $\mathbb{O}(\mathcal{D}, V_{\mathcal{D}}, \xi)$ that returns $(\widehat{\mathbb{B}}V_{\mathcal{D}}(\cdot,\cdot), \Gamma(\cdot,\cdot))$, a tuple of Empirical Bellman operator and uncertainty quantifier, satisfying*

$$\mathbb{P}_{\mathcal{D}}\Big(\big|(\widehat{\mathbb{B}}V_{\mathcal{D}})(x,a) - (\mathbb{B}_M V_{\mathcal{D}})(x,a)\big| \le \Gamma(x,a) \text{ for all } (x,a) \in \mathcal{S}\times\mathcal{A}\Big) \ge 1-\xi.$$

Remark 4 *Here we adapt a test function $V_{\mathcal{D}}$ that can depend on the dataset $\mathcal{D}$ itself. Therefore, Γ is a function that depends on both the dataset and the test function class. We do not specify the test function class in this definition, and we will discuss its specific realization in Section 5.*

Algorithm 1 Pessimistic Policy Evaluation (PPE)

Require: Offline dataset $\{\mathcal{D}_{i,h}\}_{h=1}^H$, policy $\pi = (\pi_h)_{h=1}^H$, confidence probability $\delta \in (0,1)$.

1: Initialize $\widehat{V}_{i,H+1}^{\pi}(\cdot) \leftarrow 0$, $\forall i \in [n]$.
2: **for** step $h = H, H-1, \ldots, 1$ **do**
3: Let $(\widehat{\mathbb{B}}_{i,h}\widehat{V}_{i,h+1}^{\pi})(\cdot,\cdot), \Gamma_{i,h}(\cdot,\cdot) \leftarrow \mathbb{O}(\mathcal{D}_{i,h}, \widehat{V}_{i,h+1}^{\pi}, \delta)$
4: Set $\widehat{Q}_{i,h}^{\pi}(\cdot,\cdot) \leftarrow \min\{H-h+1, (\widehat{\mathbb{B}}_{i,h}\widehat{V}_{i,h+1}^{\pi})(\cdot,\cdot) - \Gamma_{i,h}(\cdot,\cdot)\}^+$
5: Set $\widehat{V}_{i,h}^{\pi}(\cdot) \leftarrow \langle \widehat{Q}_{i,h}^{\pi}(\cdot,\cdot), \pi_h(\cdot|\cdot)\rangle_{\mathcal{A}}$
6: **end for**

Ensure: $\widehat{V}_{i,1}^{\pi}(\cdot), \ldots, \widehat{V}_{i,H}^{\pi}(\cdot), \widehat{Q}_{i,1}^{\pi}(\cdot,\cdot), \ldots, \widehat{Q}_{i,H}^{\pi}(\cdot,\cdot)$.

Based on the oracle $\mathbb{O}$, we propose our pessimistic policy evaluation (PPE) algorithm as Algorithm 1. In general, PPE takes a given policy π as its input, and its goal is to evaluate the V value and Q value $\{(V_{i,h}^{\pi}, Q_{i,h}^{\pi})\}_{h=1}^H$ of π on MDP $\mathcal{M}_i$. Since the agent is not allowed to interact with $\mathcal{M}_i$, PPE evaluates the value based on the offline dataset $\{\mathcal{D}_{i,h}\}_{h=1}^H$. At each stage h, PPE utilizes the oracle $\mathbb{O}$ and obtains the empirical Bellman operator based on $\mathcal{D}_{i,h}$ as well as its uncertainty quantifier, with high probability. Then PPE applies the *pessimism principle* to build the estimation of the Q function based on the empirical Bellman operator and the uncertainty quantifier. Such a principle has been widely studied and used in offline policy optimization, such as pessimistic value iteration (PEVI) [Jin et al., 2021]. To compare with, we use the pessimism principle in the policy evaluation problem.

4.2 Model-based approach: pessimistic empirical risk minimization

Given PPE, we propose algorithms that have the ZSG ability. We first propose a pessimistic empirical risk minimization (PERM) method which is model-based and conceptually simple. The algorithm details are in Algorithm 2. In detail, for each dataset $\mathcal{D}_i$ drawn from i-th environments, PERM builds a model using PPE to evaluate the policy π under the environment $\mathcal{M}_i$. Then PERM outputs a policy $\pi^{\text{PERM}} \in \Pi$ that maximizes the average pessimistic value, i.e., $1/n \sum_{i=1}^n \widehat{V}_{i,1}^{\pi}(x_1)$. Our approach is inspired by the classical empirical risk minimization approach adopted in supervised learning, and the Optimistic Model-based ERM proposed in Ye et al. [2023] for online RL. Our setting is more challenging than the previous ones due to the RL setting and the offline setting, where the interaction between the agent and the environment is completely disallowed. Therefore, unlike Ye et al. [2023], which adopted an optimism-style estimation to the policy value, we adopt a pessimism-style estimation to fight the distribution shift issue in the offline setting.

Next we propose a theoretical analysis of PERM. Denote $\mathcal{N}_{\epsilon}^{\Pi}$ as the ϵ-covering number of the policy space Π w.r.t. distance $\mathrm{d}(\pi^1, \pi^2) = \max_{s\in\mathcal{S}, h\in[H]} \|\pi_h^1(\cdot|s) - \pi_h^2(\cdot|s)\|_1$. Then we have the following theorem to provide an upper bound of the suboptimality gap of the output policy π^{PERM}.

Algorithm 2 Pessimistic Empirical Risk Minimization (PERM)

Require: Offline dataset $\mathcal{D} = \{\mathcal{D}_i\}_{i=1}^n, \mathcal{D}_i := \{(x_{i,h}^{\tau}, a_{i,h}^{\tau}, r_{i,h}^{\tau})_{h=1}^H\}_{\tau=1}^K$, policy class Π, confidence probability $\delta \in (0,1)$, a pessimistic offline policy evaluation algorithm **Evaluation** as a subroutine.

1: Set $\mathcal{D}_{i,h} = \{(x_{i,h}^{\tau}, a_{i,h}^{\tau}, r_{i,h}^{\tau}, x_{i,h+1}^{\tau})\}_{\tau=1}^K$
2: $\pi^{\text{PERM}} = \operatorname{argmax}_{\pi\in\Pi} \frac{1}{n}\sum_{i=1}^n \widehat{V}_{i,1}^{\pi}(x_1)$,
where $[\widehat{V}_{i,1}^{\pi}(\cdot), \cdot, \ldots, \cdot] = \textbf{Evaluation}\Big(\{\mathcal{D}_{i,h}\}_{h=1}^H, \pi, \delta/(3nH\mathcal{N}_{(Hn)^{-1}}^{\Pi})\Big)$

Ensure: π^{PERM}.

Theorem 5 *Set the Evaluation subroutine in Algorithm 2 as PPE (Algo.1). Let $\Gamma_{i,h}$ be the uncertainty quantifier returned by $\mathbb{O}$ through the PERM. Then w.p. at least $1-\delta$, the output π^{PERM} of Algorithm 2 satisfies*

$$\text{SubOpt}(\pi^{\text{PERM}}) \le \underbrace{7\sqrt{\frac{2\log(6\mathcal{N}_{(Hn)^{-1}}^{\Pi}/\delta)}{n}}}_{I_1:\text{Supervised learning (SL) error}} + \underbrace{\frac{2}{n}\sum_{i=1}^n\sum_{h=1}^H \mathbb{E}_{i,\pi^*}\left[\Gamma_{i,h}(s_h,a_h)|s_1=x_1\right]}_{I_2:\text{Reinforcement learning (RL) error}}, \tag{1}$$

where $\mathbb{E}_{i,\pi^*}$ is w.r.t. the trajectory induced by π^* with the transition $\mathcal{P}_i$ in the underlying MDP $\mathcal{M}_i$.

Proof See Appendix C.1. ■

Remark 6 *The covering number $\mathcal{N}^{\Pi}_{(Hn)^{-1}}$ depends on the policy class Π. Without any specific assumptions, the policy class Π that consists of all the policies $\pi = \{\pi_h\}_{h=1}^H, \pi_h : \mathcal{S} \mapsto \Delta(\mathcal{A})$ and the log ϵ-covering number $\log \mathcal{N}^{\Pi}_{\epsilon} = O(|\mathcal{A}||\mathcal{S}|H \log(1 + |\mathcal{A}|/\epsilon))$.*

Remark 7 *The SL error can be easily improved to a distribution-dependent bound $\log \mathcal{N} \cdot Var/\sqrt{n}$, where $\mathcal{N}$ is the covering number term denoted in I_1, $Var = \max_\pi \mathbb{V}_{c\sim C} V^\pi_{c,1}(x_1)$ is the variance of the context distribution, by using a Bernstein-type concentration inequality in our proof. Therefore, for the singleton environment case where $|C| = 1$, our suboptimality gap reduces to the one of PEVI in Jin et al. [2021].*

Theorem 5 shows that the ZSG gap of PERM is bounded by two terms I_1 and I_2. I_1, which we call it *supervised learning error*, depends on the number of environments n in the offline dataset $\mathcal{D}$ and the covering number of the function (policy) class, which is similar to the generalization error in supervised learning. I_2, which we call it *reinforcement learning error*, is decided by the optimal policy π^* that achieves the best zero-shot generalization performance and the uncertainty quantifier $\Gamma_{i,h}$. In general, I_2 is the "intrinsic uncertainty" denoted by Jin et al. [2021] over n MDPs, which characterizes how well each dataset $\mathcal{D}_i$ covers the optimal policy π^*. Since I_2 depends on the concrete form of the uncertainty quantifier, we leave the discussion of the concrete convergence guarantee to Section 5.

4.3 Model-free approach: pessimistic proximal policy optimization

Algorithm 3 Pessimistic Proximal Policy Optimzation (PPPO)

Require: Offline dataset $\mathcal{D} = \{\mathcal{D}_i\}_{i=1}^n, \mathcal{D}_i := \{(x^\tau_{i,h}, a^\tau_{i,h}, r^\tau_{i,h})_{h=1}^H\}_{\tau=1}^K$, confidence probability $\delta \in (0,1)$, a pessimistic offline policy evaluation algorithm **Evaluation** as a subroutine.

1: Set $\mathcal{D}_{i,h} = \{(x^{\tau\cdot H+h}_{i,h}, a^{\tau\cdot H+h}_{i,h}, r^{\tau\cdot H+h}_{i,h}, x^{\tau\cdot H+h}_{i,h+1})\}_{\tau=0}^{\lfloor K/H \rfloor - 1}$

2: Set $\pi_{0,h}(\cdot|\cdot)$ as uniform distribution over $\mathcal{A}$ and $\widehat{Q}^{\pi_0}_{0,h}(\cdot,\cdot)$ as zero functions.

3: **for** $i = 1, 2, \cdots, n$ **do**

4: Set $\pi_{i,h}(\cdot|\cdot) \propto \pi_{i-1,h}(\cdot|\cdot) \cdot \exp(\alpha \cdot \widehat{Q}^{\pi_{i-1}}_{i-1,h}(\cdot,\cdot))$

5: Set $[\cdot, \ldots, \cdot, \widehat{Q}^{\pi_i}_{i,1}(\cdot,\cdot), \ldots, \widehat{Q}^{\pi_i}_{i,H}(\cdot,\cdot)] =$ **Evaluation**$(\{\mathcal{D}_{i,h}\}_{h=1}^H, \pi_i, \delta/(nH))$

6: **end for**

Ensure: $\pi^{\text{PPPO}} = \text{random}(\pi_1, ..., \pi_n)$

PERM in Algorithm 2 works as a general model-based algorithm framework to enable ZSG for any pessimistic policy evaluation oracle. However, note that in order to implement PERM, one needs to maintain n different models or critic functions simultaneously in order to evaluate $\sum_{i=1}^n \widehat{V}^\pi_{i,1}(x_1)$ for any candidate policy π. Note that existing online RL [Ghosh et al., 2021] achieves ZSG by a model-free approach, which only maintains n policies rather than models/critic functions. Therefore, one natural question is whether we can design a *model-free* offline RL algorithm also with access only to policies.

We propose the pessimistic proximal policy optimization (PPPO) in Algorithm 3 to address this issue. Our algorithm is inspired by the optimistic PPO [Cai et al., 2020] originally proposed for online RL. PPPO also adapts PPE as its subroutine to evaluate any given policy pessimistically. Unlike PERM, PPPO only maintains n policies $\pi_1, ..., \pi_n$, each of them is associated with an MDP $\mathcal{M}_n$ from the offline dataset. In detail, PPPO assigns an order for MDPs in the offline dataset and names them $\mathcal{M}_1, ..., \mathcal{M}_n$. For i-th MDP $\mathcal{M}_i$, PPPO selects the i-th policy π_i as the solution of the proximal policy optimization starting from π_{i-1}, which is

$$\pi_i \leftarrow \operatorname*{argmax}_\pi V^\pi_{i-1,1}(x_1) - \alpha^{-1}\mathbb{E}_{i-1,\pi_{i-1}}[\text{KL}(\pi\|\pi_{i-1})|s_1 = x_1], \tag{2}$$

where α is the step size parameter. Since $V^{\pi}_{i-1,1}(x_1)$ is not achievable, we use a linear approximation $L_{i-1}(\pi)$ to replace $V^{\pi}_{i-1,1}(x_1)$, where

$$L_{i-1}(\pi) = V^{\pi_{i-1}}_{i-1,1}(x_1) + \mathbb{E}_{i-1,\pi_{i-1}}\left[\sum_{h=1}^{H}\langle \widehat{Q}^{\pi_{i-1}}_{i-1,h}(x_h,\cdot), \pi_h(\cdot|x_h) - \pi_{i-1,h}(\cdot|x_h)\rangle \middle| s_1 = x_1\right], \quad (3)$$

where $\widehat{Q}^{\pi_{i-1}}_{i-1,h} \approx Q^{\pi_{i-1}}_{i-1,h}$ are the Q values evaluated on the offline dataset for $\mathcal{M}_{i-1}$. (2) and (3) give us a close-form solution of π in Line 4 in Algorithm 3. Such a routine corresponds to one iteration of PPO [Schulman et al., 2017]. Finally, PPPO outputs π^{PPPO} as a random selection from $\pi_1, ..., \pi_n$.

Remark 8 *In Algorithm 3. First, we adopt a data-splitting trick [Jin et al., 2021] to build $\mathcal{D}_{i,h}$, where we only utilize each trajectory once for one data tuple at some stage h. It is only used to avoid the statistical dependency of $\widehat{V}^{\pi_i}_{i,h+1}(\cdot)$ and $x^{\tau}_{i,h+1}$ for the purpose of theoretical analysis.*

Next we have our theorem to bound the suboptimality of PPPO.

Theorem 9 *Set the Evaluation subroutine in Algorithm 3 as Algorithm 1. Let $\Gamma_{i,h}$ be the uncertainty quantifier returned by $\mathbb{O}$ through the PPPO. Selecting $\alpha = 1/\sqrt{H^2 n}$. Then selecting $\delta = 1/8$, w.p. at least $2/3$, we have*

$$\textit{SubOpt}(\pi^{\textit{PPPO}}) \le 10\Bigg(\underbrace{\sqrt{\frac{\log|\mathcal{A}|H^2}{n}}}_{I_1:\textit{SL error}} + \underbrace{\frac{1}{n}\sum_{i=1}^{n}\sum_{h=1}^{H}\mathbb{E}_{i,\pi^*}\left[\Gamma_{i,h}(s_h,a_h)|s_1=x_1\right]}_{I_2:\textit{RL error}}\Bigg).$$

where $\mathbb{E}_{i,\pi^*}$ is w.r.t. the trajectory induced by π^* with the transition $\mathcal{P}_i$ in the underlying MDP $\mathcal{M}_i$.

Proof See Appendix C.2. ■

Theorem 9 shows that the suboptimality gap of PPPO can also be bounded by the SL error I_1 and RL error I_2. Interestingly, I_1 in Theorem 9 for PPPO only depends on the cardinality of the action space $|\mathcal{A}|$, which is different from the covering number term in I_1 for PERM. Such a difference is due to the fact that PPPO outputs the final policy π^{PPPO} as a random selection from n existing policies, while PERM outputs one policy π^{PERM}. Whether these two guarantees can be unified into one remains an open question.

5 Provable generalization for offline linear MDPs

In this section, we instantiate our Algo.2 and Algo.3 for general MDPs on specific MDP classes. We consider the linear MDPs defined as follows.

Assumption 10 (Linear MDP, Yang and Wang 2019, Jin et al. 2019) *We assume $\forall i \in C, \mathcal{M}_i$ is a linear MDP with a known feature map $\phi : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^d$ if there exist d unknown measures $\mu_{i,h} = (\mu^{(1)}_{i,h}, \ldots, \mu^{(d)}_{i,h})$ over $\mathcal{S}$ and an unknown vector $\theta_{i,h} \in \mathbb{R}^d$ such that*

$$P_{i,h}(x' \,|\, x, a) = \langle \phi(x,a), \mu_{i,h}(x')\rangle, \quad \mathbb{E}\big[r_{i,h}(s_h,a_h) \,\big|\, s_h = x, a_h = a\big] = \langle \phi(x,a), \theta_{i,h}\rangle \quad (4)$$

for all $(x,a,x') \in \mathcal{S}\times\mathcal{A}\times\mathcal{S}$ at every step $h \in [H]$. We assume $\|\phi(x,a)\| \le 1$ for all $(x,a) \in \mathcal{S}\times\mathcal{A}$ and $\max\{\|\mu_{i,h}(\mathcal{S})\|, \|\theta_{i,h}\|\} \le \sqrt{d}$ at each step $h \in [H]$, and we define $\|\mu_{i,h}(\mathcal{S})\| = \int_{\mathcal{S}} \|\mu_{i,h}(x)\|\,\mathrm{d}x$.

Remark 11 *We assume that each environment $\mathcal{M}_i$ shares the same feature mapping $\phi(x,a)$. Such an assumption is for the ease of presentation, and our results can be easily extended to the setting where different environments enjoy different feature mappings.*

We first specialize the general PPE algorithm (Algo.1) to obtain the PPE algorithm tailored for linear MDPs (Algo.4). This specialization is achieved by constructing $\widehat{\mathbb{B}}_{i,h}\widehat{V}^{\pi}_{i,h+1}$, $\Gamma_{i,h}$, and $\widehat{V}^{\pi}_{i,h}$ based on the dataset $\mathcal{D}_i$. We denote the set of trajectory indexes in $\mathcal{D}_{i,h}$ as $\mathcal{B}_{i,h}$. Algo.4 subsequently functions as the policy evaluation subroutine in Algo.2 and Algo.3 for linear MDPs. In detail, we construct $\widehat{\mathbb{B}}_{i,h}\widehat{V}_{i,h+1}$ (which is the estimation of $\mathbb{B}_{i,h}\widehat{V}_{i,h+1}$) as $(\widehat{\mathbb{B}}_{i,h}\widehat{V}_{i,h+1})(x,a) = \phi(x,a)^{\top}\widehat{w}_{i,h}$, where

$$\widehat{w}_{i,h} = \text{argmin}_{w\in\mathbb{R}^d}\textstyle\sum_{\tau\in\mathcal{B}_{i,h}}\big(r^{\tau}_{i,h} + \widehat{V}_{i,h+1}(x^{-,\tau}_{i,h}) - \phi(x^{\tau}_{i,h}, a^{\tau}_{i,h})^{\top}w\big)^2 + \lambda\cdot\|w\|_2^2 \quad (5)$$

with $\lambda > 0$ being the regularization parameter. The closed-form solution to (5) is in Line 4 in Algorithm 4. Besides, we construct the uncertainty quantifier $\Gamma_{i,h}$ based on $\mathcal{D}_i$ as

$$\Gamma_{i,h}(x,a) = \beta(\delta) \cdot \|\phi(x,a)\|_{\Lambda_{i,h}^{-1}}, \Lambda_{i,h} = \sum_{\tau \in \mathcal{B}_{i,h}} \phi(x_{i,h}^\tau, a_{i,h}^\tau)\phi(x_{i,h}^\tau, a_{i,h}^\tau)^\top + \lambda \cdot I^2, \quad (6)$$

with $\beta(\delta) > 0$ being the scaling parameter.

Algorithm 4 Pessimistic Policy Evaluation (PPE): Linear MDP

Require: Offline dataset $\{\mathcal{D}_{i,h}\}_{h=1}^H, \mathcal{D}_{i,h} = \{(x_{i,h}^\tau, a_{i,h}^\tau, r_{i,h}^\tau, x_{i,h}^{-,\tau})\}_{\tau \in \mathcal{B}_{i,h}}$, policy π, confidence probability $\delta \in (0,1)$.

1: Initialize $\widehat{V}_{i,H+1}^\pi(\cdot) \leftarrow 0, \forall i \in [n]$.
2: **for** step $h = H, H-1, \ldots, 1$ **do**
3: Set $\Lambda_{i,h} \leftarrow \sum_{\tau \in \mathcal{B}_{i,h}} \phi(x_{i,h}^\tau, a_{i,h}^\tau)\phi(x_{i,h}^\tau, a_{i,h}^\tau)^\top + \lambda \cdot I$.
4: Set $\widehat{w}_{i,h} \leftarrow \Lambda_{i,h}^{-1}(\sum_{\tau \in \mathcal{B}_{i,h}} \phi(x_{i,h}^\tau, a_{i,h}^\tau) \cdot (r_{i,h}^\tau + \widehat{V}_{i,h+1}^\pi(x_{i,h}^{-,\tau})))$.
5: Set $\Gamma_{i,h}(\cdot,\cdot) \leftarrow \beta(\delta) \cdot (\phi(\cdot,\cdot)^\top \Lambda_{i,h}^{-1} \phi(\cdot,\cdot))^{1/2}$.
6: Set $\widehat{Q}_{i,h}^\pi(\cdot,\cdot) \leftarrow \min\{\phi(\cdot,\cdot)^\top \widehat{w}_{i,h} - \Gamma_{i,h}(\cdot,\cdot), H-h+1\}^+$.
7: Set $\widehat{V}_{i,h}^\pi(\cdot) \leftarrow \langle \widehat{Q}_{i,h}^\pi(\cdot,\cdot), \pi_h(\cdot|\cdot)\rangle_{\mathcal{A}}$
8: **end for**

Ensure: $\widehat{V}_{i,1}^\pi(\cdot), \ldots, \widehat{V}_{i,H}^\pi(\cdot), \widehat{Q}_{i,1}^\pi(\cdot,\cdot), \ldots, \widehat{Q}_{i,H}^\pi(\cdot,\cdot)$.

The following theorem shows the suboptimality gaps for Algo.2 (utilizing subroutine Algo.4) and Algo.3 (also with subroutine Algo.4).

Theorem 12 *Under Assumption 10, in Algorithm 4, we set* $\lambda = 1, \quad \beta(\delta) = c \cdot dH\sqrt{\log(2dHK/\delta)}$, *where* $c > 0$ *is a positive constant. Then, we have:*
(i) for the output policy π^{PERM} *of Algo.2 with subroutine Algo.4, with probability at least* $1-\delta$, *the suboptimality gap satisfies*

$$SubOpt(\pi^{PERM}) \le 7\sqrt{\frac{7\log(6\mathcal{N}_{(Hn)^{-1}}^{\Pi}/\delta)}{n}} + \frac{2\beta\big(\frac{\delta}{3nH\mathcal{N}_{(Hn)^{-1}}^{\Pi}}\big)}{n} \sum_{i=1}^n \sum_{h=1}^H \mathbb{E}_{i,\pi^*}\Big[\|\phi(s_h,a_h)\|_{\bar{\Lambda}_{i,h}^{-1}} \,\Big|\, s_1 = x_1\Big], \quad (7)$$

(ii) for the output policy π^{PPPO} *of Algo.3 with subroutine Algo.4, setting* $\delta = 1/8$, *then with probability at least* $2/3$, *the suboptimality gap satisfies*

$$\text{SubOpt}(\pi^{\text{PPPO}}) \le 10\bigg(\sqrt{\frac{\log|\mathcal{A}|H^2}{n}} + \frac{\beta\big(\frac{1}{4nH}\big)}{n} \sum_{i=1}^n \sum_{h=1}^H \mathbb{E}_{i,\pi^*}\Big[\|\phi(s_h,a_h)\|_{\bar{\Lambda}_{i,h}^{-1}} \,\Big|\, s_1 = x_1\Big]\bigg), \quad (8)$$

where $\mathbb{E}_{i,\pi^*}$ *is with respect to the trajectory induced by* π^* *with the transition* $\mathcal{P}_i$ *in the underlying MDP* $\mathcal{M}_i$ *given the fixed matrix* $\widetilde{\Lambda}_{i,h}$ *or* $\bar{\Lambda}_{i,h}$.

Proof See Appendix D.1. ■

$\|\phi(s_h,a_h)\|_{\Lambda_{i,h}^{-1}}$ indicates how well the state-action pair (s_h,a_h) is covered by the dataset $\mathcal{D}_i$. The term $\sum_{i=1}^n \sum_{h=1}^H \mathbb{E}_{i,\pi^*}\Big[\|\phi(s_h,a_h)\|_{\Lambda_{i,h}^{-1}} \,\Big|\, s_1 = x_1\Big]$ in the suboptimality gap in Theorem 12 is small if for each context $i \in [n]$, the dataset $\mathcal{D}_i$ well covers the trajectory induced by the optimal policy π^* on the corresponding MDP $\mathcal{M}_i$.

Well-explored behavior policy Next we consider a case where the dataset $\mathcal{D}$ consists of i.i.d. trajectories collecting from different environments. Suppose $\mathcal{D}$ consists of n independent datasets $\mathcal{D}_1, \ldots, \mathcal{D}_n$, and for each environment i, $\mathcal{D}_i$ consists of K trajectories $\mathcal{D}_i = \{(x_{i,h}^\tau, a_{i,h}^\tau, r_{i,h}^\tau)_{h=1}^H\}_{\tau=1}^K$ independently and identically induced by a fixed behavior policy $\bar{\pi}_i$ in the linear MDP $\mathcal{M}_i$. We have the following assumption:

[2]Spefically, for Algo.2, $\widetilde{\Lambda}_{i,h} = \sum_{\tau=1}^K \phi(x_{i,h}^\tau, a_{i,h}^\tau)\phi(x_{i,h}^\tau, a_{i,h}^\tau)^\top + \lambda \cdot I$, for Algo.3, $\bar{\Lambda}_{i,h} = \sum_{\tau=1}^{\lfloor K/H \rfloor - 1} \phi(x_{i,h}^{\tau \cdot H+h}, a_{i,h}^{\tau \cdot H+h})\phi(x_{i,h}^{\tau \cdot H+h}, a_{i,h}^{\tau \cdot H+h})^\top + \lambda \cdot I$ due to the data-splitting techniques.

Definition 13 (Well-Explored Policy, Duan et al. 2020, Jin et al. 2021) *For an behavior policy $\bar{\pi}$ and an episodic linear MDP $\mathcal{M}$ with the feature mapping ϕ, we say $\bar{\pi}$ well-explores $\mathcal{M}$ with constant c if there exists an absolute positive constant $c > 0$ such that*

$$\forall h \in [H], \lambda_{\min}(\Sigma_h) \geq c/d, \quad \textit{where } \Sigma_h = \mathbb{E}_{\bar{\pi},\mathcal{M}}\big[\phi(s_h, a_h)\phi(s_h, a_h)^\top\big].$$

A well-explored policy guarantees that the obtained trajectories is "uniform" enough to represent any policy and value function. The following corollary shows that with the above assumption, the suboptimality gaps of Algo.2 (with subroutine Algo.4) and Algo.3 (with subroutine Algo.4) decay to 0 when n and K are large enough.

Corollary 14 *Suppose that for each $i \in [n]$, $\mathcal{D}_i$ is generated by behavior policy $\bar{\pi}_i$ which well-explores MDP $\mathcal{M}_i$ with constant $c_i \geq c_{min}$. In Algo.4, we set $\lambda = 1, \beta(\delta) = c' \cdot dH\sqrt{\log(4dHK/\delta)}$ where $c' > 0$ is a positive constant. Suppose we have $K \geq 40d/c_{min}\log(4dnH/\delta)$ and set $C_n^* := 1/n \cdot \sum_{i=1}^n c_i^{-1/2}$. Then we have:*
(i) for the output π^{PERM} of Algo.2 with subroutine Algo.4, with probability at least $1-\delta$, the suboptimality gap satisfies

$$\textit{SubOpt}(\pi^{PERM}) \leq 7\sqrt{\frac{2\log(6\mathcal{N}^{\Pi}_{(Hn)^{-1}}/\delta)}{n}} + 2\sqrt{2}c' \cdot d^{3/2}H^2K^{-1/2}\sqrt{\log(12dHnK\mathcal{N}^{\Pi}_{(Hn)^{-1}}/\delta)} \cdot C_n^*, \tag{9}$$

(ii) for the output policy π^{PPPO} of Algo.3 with subroutine Algo.4, setting $\delta = 1/8$, then with probability at least $2/3$, the suboptimality gap satisfies

$$\text{SubOpt}(\pi^{\text{PPPO}}) \leq 10\bigg(\sqrt{\frac{\log|\mathcal{A}|H^2}{n}} + 2\sqrt{2}c' \cdot d^{3/2}H^{2.5}K^{-1/2}\sqrt{\log(16dHnK)} \cdot C_n^*\bigg). \tag{10}$$

Proof See Appendix D.2. ■

Remark 15 *The mixed coverage parameter $C_n^* = \frac{1}{n}\sum_{i=1}^n \frac{1}{\sqrt{c_i}}$ is small if for any $i \in [n]$, c_i is large,* i.e.*, the minimum eigenvalue of $\Sigma_{i,h} = \mathbb{E}_{\bar{\pi}_i,\mathcal{M}_i}\big[\phi(s_h, a_h)\phi(s_h, a_h)^\top\big]$ is large. Note that $\lambda_{min}(\Sigma_{i,h})$ indicates how well the behavior policy $\bar{\pi}_i$ explores the state-action pairs on MDP $\mathcal{M}_i$; this shows that if for each environment $i \in [n]$, the behavior policy explores $\mathcal{M}_i$ well, the suboptimality gap will be small.*

Remark 16 *Under the same conditions of Corollary 14, we have:*
(i) If $n \geq \frac{392\log(6\mathcal{N}^{\Pi}_{(Hn)^{-1}}/\delta)}{\epsilon^2}$ and $K \geq \max\{\frac{40d}{c_{min}}\log(\frac{4dnH}{\delta}), \frac{32c'^2d^3H^4\log(12dHnK\mathcal{N}^{\Pi}_{(Hn)^{-1}}/\delta)C_n^{*2}}{\epsilon^2}\}$, then w.p. at least $1-\delta$, SubOpt$(\pi^{PERM}) \leq \epsilon$.
(ii) If $n \geq \frac{400H^2\log(|\mathcal{A}|)}{\epsilon^2}$ and $K \geq \max\{\frac{40d}{c_{min}}\log(16dnH), \frac{32c'^2d^3H^5\log(16dHnK)C_n^{*2}}{\epsilon^2}\}$, then w.p. at least $2/3$, SubOpt$(\pi^{PPPO}) \leq \epsilon$.

Corollary 14 suggests that both of our proposed algorithms enjoy the $O(n^{-1/2} + K^{-1/2} \cdot C_n^*)$ convergence rate to the optimal policy π^* given a well-exploration data collection assumption, where C_n^* is a mixed coverage parameter over n environments defined in Corollary 14.

6 Conclusion and Future Work

In this work, we study the zero-shot generalization (ZSG) performance of offline reinforcement learning (RL). We propose two offline RL frameworks, pessimistic empirical risk minimization and pessimistic proximal policy optimization, and show that both of them can find the optimal policy with ZSG ability. We also show that such a generalization property does not hold for offline RL without knowing the context information of the environment, which demonstrates the necessity of our proposed new algorithms. Currently, our theorems and algorithm design depend on the i.i.d. assumption of the environment selection. How to relax such an assumption remains an interesting future direction.

References

Alekh Agarwal, Yuda Song, Wen Sun, Kaiwen Wang, Mengdi Wang, and Xuezhou Zhang. Provable benefits of representational transfer in reinforcement learning. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 2114–2187. PMLR, 2023.

Anurag Ajay, Ge Yang, Ofir Nachum, and Pulkit Agrawal. Understanding the generalization gap in visual reinforcement learning. 2021.

Joshua Albrecht, Abraham Fetterman, Bryden Fogelman, Ellie Kitanidis, Bartosz Wróblewski, Nicole Seo, Michael Rosenthal, Maksis Knutins, Zack Polizzi, James Simon, et al. Avalon: A benchmark for rl generalization using procedurally generated worlds. *Advances in Neural Information Processing Systems*, 35:12813–12825, 2022.

Chenjia Bai, Lingxiao Wang, Zhuoran Yang, Zhihong Deng, Animesh Garg, Peng Liu, and Zhaoran Wang. Pessimistic bootstrapping for uncertainty-driven offline reinforcement learning. *arXiv preprint arXiv:2202.11566*, 2022.

Emmanuel Bengio, Joelle Pineau, and Doina Precup. Interference and generalization in temporal difference learning. *International Conference On Machine Learning*, 2020.

Martin Bertran, Natalia Martinez, Mariano Phielipp, and Guillermo Sapiro. Instance-based generalization in reinforcement learning. *Advances in Neural Information Processing Systems*, 33: 11333–11344, 2020.

Avinandan Bose, Simon Shaolei Du, and Maryam Fazel. Offline multi-task transfer rl with representational penalization. *arXiv preprint arXiv:2402.12570*, 2024.

Emma Brunskill and Lihong Li. Sample complexity of multi-task reinforcement learning. *arXiv preprint arXiv:1309.6821*, 2013.

Qi Cai, Zhuoran Yang, Chi Jin, and Zhaoran Wang. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning*, pages 1283–1294. PMLR, 2020.

Karl Cobbe, Oleg Klimov, Christopher Hesse, Taehoon Kim, and J. Schulman. Quantifying generalization in reinforcement learning. *International Conference On Machine Learning*, 2018.

Karl Cobbe, Chris Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning. In *International conference on machine learning*, pages 2048–2056. PMLR, 2020.

Yaqi Duan, Zeyu Jia, and Mengdi Wang. Minimax-optimal off-policy evaluation with linear function approximation. In *International Conference on Machine Learning*, pages 2701–2709. PMLR, 2020.

Andy Ehrenberg, Robert Kirk, Minqi Jiang, Edward Grefenstette, and Tim Rocktäschel. A study of off-policy learning in environments with procedural content generation. In *ICLR Workshop on Agent Learning in Open-Endedness*, 2022.

Damien Ernst, Pierre Geurts, and Louis Wehenkel. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6, 2005.

Kevin Frans and Phillip Isola. Powderworld: A platform for understanding generalization via rich task distributions. *arXiv preprint arXiv:2211.13051*, 2022.

Kamyar Ghasemipour, Shixiang Shane Gu, and Ofir Nachum. Why so pessimistic? estimating uncertainties for offline rl through ensembles, and why their independence matters. *Advances in Neural Information Processing Systems*, 35:18267–18281, 2022.

Dibya Ghosh, Jad Rahme, Aviral Kumar, Amy Zhang, Ryan P Adams, and Sergey Levine. Why generalization in rl is difficult: Epistemic pomdps and implicit partial observability. *Advances in neural information processing systems*, 34:25502–25515, 2021.

Jiachen Hu, Xiaoyu Chen, Chi Jin, Lihong Li, and Liwei Wang. Near-optimal representation learning for linear bandits and linear rl. In *International Conference on Machine Learning*, pages 4349–4358. PMLR, 2021.

Haque Ishfaq, Thanh Nguyen-Tang, Songtao Feng, Raman Arora, Mengdi Wang, Ming Yin, and Doina Precup. Offline multitask representation learning for reinforcement learning. *arXiv preprint arXiv:2403.11574*, 2024.

Yiding Jiang, J Zico Kolter, and Roberta Raileanu. Uncertainty-driven exploration for generalization in reinforcement learning. In *Deep Reinforcement Learning Workshop NeurIPS 2022*.

Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. Provably efficient reinforcement learning with linear function approximation. *arXiv preprint arXiv:1907.05388*, 2019.

Ying Jin, Zhuoran Yang, and Zhaoran Wang. Is pessimism provably efficient for offline rl? In *International Conference on Machine Learning*, pages 5084–5096. PMLR, 2021.

Arthur Juliani, Ahmed Khalifa, Vincent-Pierre Berges, Jonathan Harper, Ervin Teng, Hunter Henry, Adam Crespi, Julian Togelius, and Danny Lange. Obstacle Tower: A Generalization Challenge in Vision, Control, and Planning. In *IJCAI*, 2019.

Niels Justesen, Ruben Rodriguez Torrado, Philip Bontrager, Ahmed Khalifa, Julian Togelius, and Sebastian Risi. Illuminating generalization in deep reinforcement learning through procedural level generation. *arXiv: Learning*, 2018.

Robert Kirk, Amy Zhang, Edward Grefenstette, and Tim Rocktäschel. A survey of zero-shot generalisation in deep reinforcement learning. *Journal of Artificial Intelligence Research*, 76: 201–264, 2023.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*, 2021.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 33:1179–1191, 2020.

Heinrich Küttler, Nantas Nardelli, Alexander H. Miller, Roberta Raileanu, Marco Selvatici, Edward Grefenstette, and Tim Rocktäschel. The NetHack Learning Environment. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

Sascha Lange, Thomas Gabel, and Martin Riedmiller. Batch reinforcement learning. In *Reinforcement learning: State-of-the-art*, pages 45–73. Springer, 2012.

Kimin Lee, Kibok Lee, Jinwoo Shin, and Honglak Lee. Network randomization: A simple technique for generalization in deep reinforcement learning. In *International Conference on Learning Representations. https://openreview. net/forum*, 2020.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Yao Liu, Adith Swaminathan, Alekh Agarwal, and Emma Brunskill. Provably good batch off-policy reinforcement learning without great exploration. *Advances in neural information processing systems*, 33:1264–1274, 2020.

Rui Lu, Gao Huang, and Simon S Du. On the power of multitask representation learning in linear mdp. *arXiv preprint arXiv:2106.08053*, 2021.

Clare Lyle, Mark Rowland, Will Dabney, Marta Kwiatkowska, and Yarin Gal. Learning dynamics and generalization in deep reinforcement learning. In *International Conference on Machine Learning*, pages 14560–14581. PMLR, 2022.

Marlos C. Machado, Marc G. Bellemare, Erik Talvitie, Joel Veness, Matthew J. Hausknecht, and Michael H. Bowling. Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. In *IJCAI*, 2018.

Dhruv Malik, Yuanzhi Li, and Pradeep Ravikumar. When is generalizable reinforcement learning tractable? *Advances in Neural Information Processing Systems*, 34, 2021.

Bogdan Mazoure, Ilya Kostrikov, Ofir Nachum, and Jonathan J Tompson. Improving zero-shot generalization in offline reinforcement learning using generalized similarity functions. *Advances in Neural Information Processing Systems*, 35:25088–25101, 2022.

Ishita Mediratta, Qingfei You, Minqi Jiang, and Roberta Raileanu. The generalization gap in offline reinforcement learning. *arXiv preprint arXiv:2312.05742*, 2023.

Thanh Nguyen-Tang and Raman Arora. On sample-efficient offline reinforcement learning: Data diversity, posterior sampling and beyond. *Advances in Neural Information Processing Systems*, 36, 2024.

Alex Nichol, V. Pfau, Christopher Hesse, O. Klimov, and John Schulman. Gotta learn fast: A new benchmark for generalization in rl. *ArXiv*, abs/1804.03720, 2018.

Charles Packer, Katelyn Gao, Jernej Kos, Philipp Krähenbühl, Vladlen Koltun, and Dawn Song. Assessing generalization in deep reinforcement learning. *ICLR*, 2019.

Aravind Rajeswaran, Kendall Lowrey, Emanuel Todorov, and Sham M. Kakade. Towards generalization and simplicity in continuous control. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6550–6561, 2017.

Paria Rashidinejad, Banghua Zhu, Cong Ma, Jiantao Jiao, and Stuart Russell. Bridging offline reinforcement learning and imitation learning: A tale of pessimism. *Advances in Neural Information Processing Systems*, 34:11702–11716, 2021.

Shideh Rezaeifar, Robert Dadashi, Nino Vieillard, Léonard Hussenot, Olivier Bachem, Olivier Pietquin, and Matthieu Geist. Offline reinforcement learning as anti-exploration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8106–8114, 2022.

Martin Riedmiller. Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method. In *Machine Learning: ECML 2005: 16th European Conference on Machine Learning, Porto, Portugal, October 3-7, 2005. Proceedings 16*, pages 317–328. Springer, 2005.

Mikayel Samvelyan, Robert Kirk, Vitaly Kurin, Jack Parker-Holder, Minqi Jiang, Eric Hambro, Fabio Petroni, Heinrich Küttler, Edward Grefenstette, and Tim Rocktäschel. Minihack the planet: A sandbox for open-ended reinforcement learning research. *arXiv preprint arXiv:2109.13202*, 2021.

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

Xingyou Song, Yiding Jiang, Stephen Tu, Yilun Du, and Behnam Neyshabur. Observational overfitting in reinforcement learning. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=HJli2hNKDH`.

Andrea Tirinzoni, Riccardo Poiani, and Marcello Restelli. Sequential transfer in reinforcement learning with a generative model. In *International Conference on Machine Learning*, pages 9481–9492. PMLR, 2020.

Masatoshi Uehara and Wen Sun. Pessimistic model-based offline reinforcement learning under partial coverage. *arXiv preprint arXiv:2107.06226*, 2021.

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

Huan Wang, Stephan Zheng, Caiming Xiong, and Richard Socher. On the generalization gap in reparameterizable reinforcement learning. In *International Conference on Machine Learning*, pages 6648–6658. PMLR, 2019.

Yue Wu, Shuangfei Zhai, Nitish Srivastava, Joshua Susskind, Jian Zhang, Ruslan Salakhutdinov, and Hanlin Goh. Uncertainty weighted actor-critic for offline reinforcement learning. *arXiv preprint arXiv:2105.08140*, 2021.

Tengyang Xie, Ching-An Cheng, Nan Jiang, Paul Mineiro, and Alekh Agarwal. Bellman-consistent pessimism for offline reinforcement learning. *Advances in neural information processing systems*, 34:6683–6694, 2021a.

Tengyang Xie, Nan Jiang, Huan Wang, Caiming Xiong, and Yu Bai. Policy finetuning: Bridging sample-efficient offline and online reinforcement learning. *Advances in neural information processing systems*, 34:27395–27407, 2021b.

Yuling Yan, Gen Li, Yuxin Chen, and Jianqing Fan. The efficacy of pessimism in asynchronous q-learning. *IEEE Transactions on Information Theory*, 2023.

Lin Yang and Mengdi Wang. Sample-optimal parametric q-learning using linearly additive features. In *International Conference on Machine Learning*, pages 6995–7004, 2019.

Rui Yang, Lin Yong, Xiaoteng Ma, Hao Hu, Chongjie Zhang, and Tong Zhang. What is essential for unseen goal generalization of offline goal-conditioned rl? In *International Conference on Machine Learning*, pages 39543–39571. PMLR, 2023.

Chang Ye, Ahmed Khalifa, Philip Bontrager, and Julian Togelius. Rotation, translation, and cropping for zero-shot generalization. In *2020 IEEE Conference on Games (CoG)*, pages 57–64. IEEE, 2020.

Haotian Ye, Xiaoyu Chen, Liwei Wang, and Simon Shaolei Du. On the power of pre-training for generalization in rl: provable benefits and hardness. In *International Conference on Machine Learning*, pages 39770–39800. PMLR, 2023.

Ming Yin, Yaqi Duan, Mengdi Wang, and Yu-Xiang Wang. Near-optimal offline reinforcement learning with linear representation: Leveraging variance information with pessimism. *arXiv preprint arXiv:2203.05804*, 2022.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *Advances in Neural Information Processing Systems*, 33:14129–14142, 2020.

Andrea Zanette, Martin J Wainwright, and Emma Brunskill. Provable benefits of actor-critic methods for offline reinforcement learning. *Advances in neural information processing systems*, 34:13626–13640, 2021.

Amy Zhang, Nicolas Ballas, and Joelle Pineau. A dissection of overfitting and generalization in continuous reinforcement learning. *ArXiv*, abs/1806.07937, 2018a.

Chicheng Zhang and Zhi Wang. Provably efficient multi-task reinforcement learning with model transfer. *Advances in Neural Information Processing Systems*, 34, 2021.

Chiyuan Zhang, Oriol Vinyals, Rémi Munos, and Samy Bengio. A study on overfitting in deep reinforcement learning. *ArXiv*, abs/1804.06893, 2018b.

Weitong Zhang, Jiafan He, Dongruo Zhou, Amy Zhang, and Quanquan Gu. Provably efficient representation selection in low-rank markov decision processes: from online to offline rl. In *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, pages 2488–2497, 2023.

A Offline RL without context information

In Section 4 and 5, we have proposed provable offline RL with small generalization gaps. Both PERM and PPPO require the agent to maintain policies/models for different MDPs separately. One may ask whether directly applying existing offline RL algorithms over datasets from multiple environments *without* maintaining their identity information can yield the same ZSG property. In this section, we show that such a strategy does not work, which is aligned with the existing observation of the poor generalization performance of offline RL [Mediratta et al., 2023]. In detail, given contextual MDPs $\mathcal{M}_1, ..., \mathcal{M}_n$ and their corresponding offline datasets $\mathcal{D}_1, ..., \mathcal{D}_n$, we assume the agent only has the access to the offline dataset $\bar{\mathcal{D}} = \cup_{i=1}^n \mathcal{D}_i$, where $\bar{\mathcal{D}} = \{(x^\tau_{c_\tau,h}, a^\tau_{c_\tau,h}, r^\tau_{c_\tau,h})_{h=1}^H\}_{\tau=1}^K$. Here $c_\tau \in C$ is the context information of trajectory τ, which is *unknown* to the agent.

Numerical experiments To show that the context information is important to ZSG property, we conduct numerical experiments on Combination Lock environment [Bose et al., 2024] to compare the performance of Pessimistic Value Iteration (PEVI) (Jin et al. [2021]), a representative algorithm for offline RL, and PPPO. Due to the space limit, the detailed description of experiments and results are deferred to Appendix B. Our results show that PPPO performs better, which backs up our theoretical claim.

Theoretical justification To explain why offline RL without knowing context information performs worse, we have the following proposition suggesting the offline dataset from multiple MDPs is not distinguishable from an "average MDP" if the offline dataset does not contain context information.

Proposition 17 *$\bar{\mathcal{D}}$ is compliant with* average MDP $\bar{\mathcal{M}} := \{\bar{M}_h\}_{h=1}^H$, $\bar{M}_h := \big(\mathcal{S}, \mathcal{A}, H, \bar{P}_h, \bar{r}_h\big)$,

$\bar{P}_h(x'|x,a) := \mathbb{E}_{c\sim C} \dfrac{P_{c,h}(x'|x,a)\mu_{c,h}(x,a)}{\mathbb{E}_{c\sim C}\mu_{c,h}(x,a)}$, $\mathbb{P}(\bar{r}_h = r|x,a) := \mathbb{E}_{c\sim C} \dfrac{\mathbb{P}(\bar{r}_{c,h} = r|x,a)\mu_{c,h}(x,a)}{\mathbb{E}_{c\sim C}\mu_{c,h}(x,a)}$,

where $\mu_{c,h}(\cdot,\cdot)$ is the data collection distribution of (s,a) at stage h in dataset $\mathcal{D}_c$.

Proof See Appendix E.1. ■

Proposition 17 suggests that if no context information is revealed, then the merged offline dataset $\bar{\mathcal{D}}$ is equivalent to a dataset collected from the average MDP $\bar{\mathcal{M}}$. Therefore, the output policy of existing offline algorithms like PEVI converges to the optimal policy $\bar{\pi}^*$ of the average MDP $\bar{\mathcal{M}}$. We leave the convergence analysis of PEVI to Appendix E.2. In general, $\bar{\pi}^*$ can be very different from π^* when the transition probability functions of each environment are different. For example, consider the 2-context cMDP problem shown in Figure 1, each context consists of one state and three possible actions. The offline dataset distributions μ are marked on the arrows that both of the distributions are following near-optimal policy. By Proposition 17, in average MDP $\bar{\mathcal{M}}$ the reward of the middle action is deterministically 0, while both upper and lower actions are deterministically 1. As a result, the optimal policy $\bar{\pi}^*$ will only have positive probabilities toward upper and lower actions. This leads to $\mathbb{E}_{c\sim C}[V_{c,1}^{\bar{\pi}^*}(x_1)] = 0$, though we can see that π^* is deterministically choosing the middle action and $\mathbb{E}_{c\sim C}[V_{c,1}^{\pi^*}(x_1)] = 0.5$. This theoretically illustrates that the generalization ability of offline RL algorithms without leveraging context information is weak. In a sharp contrast, imitation learning such as behavior cloning (BC) converges to the teacher policy that is independent of the specific MDP. Therefore, offline RL methods such as CQL [Kumar et al., 2020] might enjoy worse generalization performance compared with BC, which aligns with the observation made by Mediratta et al. [2023].

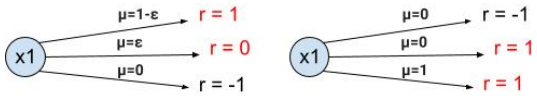

Figure 1: A cMDP problem with 2 equally distributed contexts.

B Experiment

In this section, we run experiments on both synthetic data and real-world data to validate the effectiveness of our proposed algorithm. The main idea of our proposed PERM and PPPO is to

leverage multiple environments individually, unlike previous approaches which essentially treat different environments as the same one. Therefore, we will mainly verify how the different number of contexts (Appendix B) and the number of value networks (Appendix B.2) affect the performance of the baseline algorithm.

B.1 Experiments on synthetic data

Environments We are using the comblock framework that we adapted from [Bose et al., 2024], where we directly record the policies and Q-values through state-action pairs instead of recovering them linear-algebraically through rich observations. We leverage the PyTorch framework with CPU device to process the tabular numerical operations. The entire dataset generation and experiment process is conducted on MacBook Pro with M3 Max chip where the dataset generation process takes about 2 hours.

We consider the Combination Lock environment adapted from [Bose et al., 2024]. At each timestep h, there are three states $s_{0,h}, s_{1,h}, s_{2,h}$ with 5 possible actions; only $s_{0,h}, s_{1,h}$ are considered as desirable states that are reachable toward final reward. The environment uniformly and independently samples 1 out of 5 actions for each desirable state $a_{0,h}, a_{1,h}$ at each timestep, where taking these actions will result transition to one of the good states $s_{0,h+1}, s_{1,h+1}$ with equal probability, otherwise the transition will deterministically to the bad state $s_{2,h+1}$ and remains in the bad states for the rest of the horizon. If the agent is staying in the good states at the end of the horizon, the reward will be 1; otherwise the agent has 0.5 probability to receive a 0.1 reward.

Implementation details We consider two experiment settings, one with 5 context environments and the other with 10 context environments. Each context environment is generated randomly. For the generation of the offline dataset, as in [Bose et al., 2024], we adopt the Exploratory Policy Search (EPS) algorithm proposed by [Agarwal et al., 2023] to obtain exploratory polices (not necessarily optimal) that cover as much of the feature space. For each context environment, 500 exploratory trajectories are i.i.d. sampled. We compare our proposed PPPO with the previous baseline PEVI Jin et al. [2021] w.r.t. their average reward. In our experiment, we calibrate the $\beta(\delta)$ parameter for PEVI to reflect the optimal performance, as well as the $\beta(\delta), \alpha$ parameters for PPPO to reflect near-optimal performance.

Experiment results We find that PPPO generally outperforms PEVI on average rewards in both contextual settings as shown in Table 2, this validates our theory hypothesis (see discussion in Appendix E.2 for an analysis of PEVI).

Number of Contexts	PEVI [Jin et al., 2021]	PPPO
5	0.0628	0.0670 ± 0.0141
10	0.0514	0.0650 ± 0.0173

Table 2: The average rewards for PEVI and PPPO algorithms in two different contextual settings with 5 and 10 contexts. In PPPO, noting that the result policy is randomly sampled from n polices, we are taking the average value and calculating the standard deviation (reported as 1-sigma error bars) of the evaluation results for trained policies $\pi_1, \ldots, \pi_n$.

B.2 Experiments on real data

Environments We conduct an extensive evaluation over the Procgen Benchmark Cobbe et al. [2020]. Our experiment is conducted on a server with NVIDIA RTX A6000 GPUs. For the offline setting of Procgen, we adopt the offline dataset collected from Mediratta et al. [2023], which includes the Procgen **expert dataset with 1M transitions** and the Procgen **mixed expert-suboptimal dataset with 1M transitions**, both of them are for the easy difficulty of the Procgen games. Following Mediratta et al. [2023], for each game in Procgen, the data is collected from 200 different Procgen levels for offline training, validate the hyperparameters online to perform model selection on the another 50 levels, and evaluate the agents' online performance on the remaining levels.

Implementation details For conceptual simplicity, we adopt the Implicit Q-Learning (IQL) framework Kostrikov et al. [2021], while simultaneously leveraging n critic value networks for different environments (see Algorithm 5, IQL-nV). Note that this isn't exactly the same optimization objective

as we proposed in Algorithm 2, but nonetheless a first-order approximation of what could be achieved with PERM framework. In our implementation of IQL-nV, we adapt the original implementation of IQL Kostrikov et al. [2021] to the one with several value networks V_{ψ_i}, the expectile optimization objectives $L_V(\psi_i)$ are the same as Kostrikov et al. [2021], while we use multiple MSE losses for Q network, with one for each V_{ψ_i} , $L_Q^i(\theta) = \mathbb{E}_{(s,a,s')\sim\mathcal{D}_i}[(r(s,a)+\gamma V_{\psi_i}(s') - Q_\theta(s,a))^2]$. Algorithm 5 gives our modified TD learning process from IQL. For the policy extraction method, we use AWR, which is the same as IQL.

Algorithm 5 TD Learning (IQL-nV)

Initialize parameters $\{\psi_i\}, \theta, \widehat{\theta}, \phi$.
for each gradient step **do**
  for each i **do**
    $\psi_i \leftarrow \psi_i - \lambda_V \nabla_{\psi_i} L_V(\psi_i)$, $\theta \leftarrow \theta - \lambda_Q \nabla_\theta L_Q^i(\theta)$
  end for
  $\widehat{\theta} \leftarrow (1-\alpha)\widehat{\theta} + \alpha\theta$
end for

In the Procgen datasets for evaluation, there are 200 randomly sampled training environments in each game, thus for practical reason we regard every 50 of them as one environment and only leverage 4 different value networks in our IQL-4V implementation. We tune the hyperparameters separately for expert and mixed expert-suboptimal datasets, which aligns with the practice of Mediratta et al. [2023]. Moreover, we found that the policy extraction via Soft Actor-Critic (SAC) implementation endows IQL-4V's policy a stochastic nature in some settings, so we tune the policy sampling method to leverage this effect. Table 3 shows our hyperparameters.

Experiment results We report the mean and standard deviation of returns of each algorithm and each task in Table 4. The results of BC and IQL are reported by Mediratta et al. [2023]. To summarize results over all tasks, for each algorithm, we also report the mean and median of the min-max normalized returns of each game. The min-max normalized return is calculated by using the r_{min} and r_{max} value for each game provided by the Procgen benchmark Cobbe et al. [2020]. From Table 4 we found that IQL-nV's min-max normalized returns across the set of Procgen games outperforms the baseline IQL, which suggests the effectiveness of our proposed approach. From another perspective, we list the difference between IQL-4V and IQL w.r.t. to different games, and we arrange the order of games based on the performance of IQL. The results for both expert and mixed expert-suboptimal datasets are recorded in Figure 2. From Figure 2, we can see that IQL-4V improves IQL mainly from the games where IQL performs bad in the sense of min-max normalized returns. This is aligned with the idea we improve the ZSG performance of offline RL, since our algorithm aims to improve the average performance over all games, rather than only a subset of them. It is worth noting that our approach still lags behind BC, which is aligned with the observation made by Mediratta et al. [2023]. We aim to develop offline RL with good ZSG performance that outperforms both BC and offline RL in the future work.

In our ablation study, we aim to verify the effectiveness of multiple value networks and the stochastic policy on the **Miner** game with the 1M expert dataset. We test IQL-nV with $n = 1, 2, 4, 8$ with stochastic policy. The results are in Table 5. We can see that by increasing the number of value networks, the performance of IQL also increases. Meanwhile, it is worth noting by using the stochastic policy also helps the performance. We hypothesize it is because that the 1M Expert Dataset of Procgen games enjoys a higher diversity w.r.t. to the action selection, which causes the performance gain brought by the use of stochastic policy.

Hyperparameter	IQL-4V (Expert)	IQL-4V (Mixed)
Learning Rate	0.0005	0.0005
Target model Weight Update	Polyak	Polyak
Batch Size	512	512
τ	0.005	0.005
Target update frequency	100	100
Temperature	3.0	3.0
Expectile	0.8	0.8
Policy Sampling	**Stochastic**	**Deterministic**

Table 3: Hyperparameters in our experiment.

Procgen game	BC(Expert)	IQL(Expert)	**IQL-4V(Expert)**	BC(Mixed)	IQL(Mixed)	**IQL-4V(Mixed)**
bigfish	4.38 ± 0.38	4.85 ± 0.52	2.72 ± 1.23	2.89 ± 0.15	4.14 ± 0.54	5.46 ± 3.03
bossfight	5.87 ± 0.26	7.62 ± 0.33	5.74 ± 1.05	5.13 ± 0.14	7.12 ± 0.43	6.8 ± 0.26
caveflyer	4.92 ± 0.28	3.43 ± 0.22	3.52 ± 1.46	4.05 ± 0.24	1.66 ± 0.67	3.2 ± 1.33
chaser	4.62 ± 0.36	3.17 ± 0.17	4.35 ± 0.55	3.43 ± 0.22	1.41 ± 0.6	1.32 ± 0.22
climber	4.91 ± 0.22	2.33 ± 0.33	3.92 ± 1.41	4.64 ± 0.29	0.57 ± 0.35	1.5 ± 0.80
miner	7.85 ± 0.32	1.66 ± 0.17	6.36 ± 1.85	6.56 ± 0.09	0.8 ± 0.1	1.64 ± 0.86
coinrun	8.26 ± 0.19	7.74 ± 0.21	9.8 ± 0.40	7.77 ± 0.24	6.0 ± 0.36	7.2 ± 1.17
dodgeball	0.98 ± 0.07	0.93 ± 0.12	1.0 ± 0.75	1.19 ± 0.14	0.87 ± 0.11	1.32 ± 0.48
fruitbot	21.18 ± 0.62	25.22 ± 0.94	22.24 ± 3.56	18.84 ± 0.7	22.0 ± 0.43	23.56 ± 4.43
heist	2.42 ± 0.14	0.58 ± 0.26	4.4 ± 0.80	2.37 ± 0.3	0.27 ± 0.03	0.6 ± 0.49
jumper	5.68 ± 0.18	4.06 ± 0.21	6.2 ± 1.17	4.63 ± 0.47	3.0 ± 0.5	4.2 ± 0.98
leaper	2.84 ± 0.07	2.44 ± 0.21	3.0 ± 1.41	2.6 ± 0.25	2.27 ± 0.53	3.6 ± 0.80
maze	4.46 ± 0.16	2.68 ± 0.31	5.0 ± 1.26	4.77 ± 0.32	2.1 ± 0.15	3.0 ± 0.63
ninja	5.88 ± 0.3	4.38 ± 0.12	6.0 ± 1.10	5.23 ± 0.12	3.23 ± 0.81	5.0 ± 1.41
plunder	4.94 ± 0.13	4.03 ± 0.14	5.38 ± 0.94	4.59 ± 0.16	3.86 ± 0.25	3.58 ± 0.83
starpilot	17.69 ± 0.59	22.88 ± 0.59	13.88 ± 3.35	17.93 ± 0.32	19.64 ± 1.79	11.72 ± 4.31
Mean	0.240	0.114	0.263	0.189	0.010	0.096
Median	0.261	0.065	0.183	0.234	-0.031	0.076

Table 4: Test performance of IQL-4V against BC and IQL baselines reported by Mediratta et al. [2023], on the **1M Expert Dataset** and **1M Mixed Expert-Suboptimal Dataset** respectively. For the IQL-4V returns, mean and standard deviation over 5 random seeds are reported.

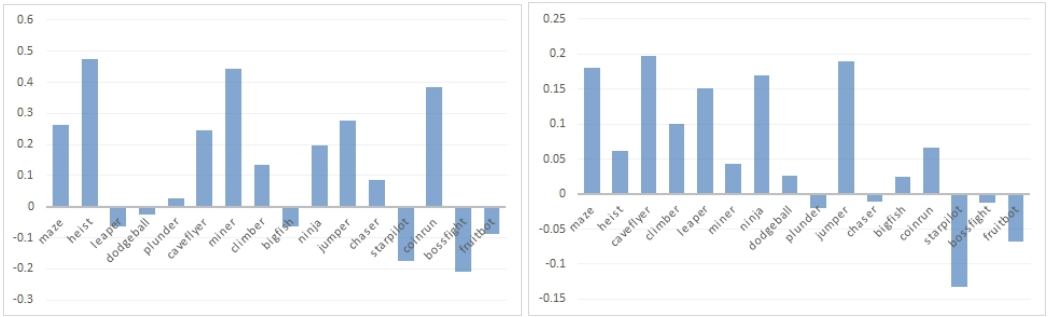

Figure 2: Differences of the mean **min-max normalized** test returns on Procgen **1M Expert Dataset (left)** and **1M Mixed Expert-Suboptimal Dataset (right)** over 5 random seeds.

Procgen game	8V-SP(Expert)	4V-SP(Expert)	2V-SP(Expert)	1V-SP(Expert)	1V-DP(Expert)
miner	7.88 ± 0.71	6.36 ± 1.85	6.85 ± 0.92	5.6 ± 1.89	1.66 ± 0.17

Table 5: The IQL-nV **ablation study** results conducted on Miner, with 1M expert dataset tested. nV-SP represents IQL-nV with a stochastic policy variant, where 1V-DP represents the default IQL setup, recorded in Mediratta et al. [2023]. Mean and standard deviation of test performance over 5 random seeds are reported.

C Proof of Theorems in Section 4

C.1 Proof of Theorem 5

We define the model estimation error as

$$\iota^{\pi}_{i,h}(x,a) = (\mathbb{B}_{i,h}\widehat{V}^{\pi}_{i,h+1})(x,a) - \widehat{Q}^{\pi}_{i,h}(x,a). \tag{11}$$

And we define the following condition

$$\left|(\widehat{\mathbb{B}}_{i,h}\widehat{V}^{\pi}_{i,h+1})(x,a) - (\mathbb{B}_{i,h}\widehat{V}^{\pi}_{i,h+1})(x,a)\right| \le \Gamma_{i,h}(x,a) \text{ for all } i \in [n], \pi \in \Pi, (x,a) \in \mathcal{S}\times\mathcal{A}, h \in [H]\,. \tag{12}$$

We introduce the following lemma to bound the model estimation error.

Lemma 18 (Model estimation error bound (Adapted from Lemma 5.1 in Jin et al. [2021]) *Under the condition of Eq.(12), we have*

$$0 \le \iota^{\pi}_{i,h}(x,a) \le 2\Gamma_{i,h}(x,a), \quad \textit{for all } \ i \in [n],\ \pi \in \Pi,\ (x,a) \in \mathcal{S}\times\mathcal{A},\ h \in [H]. \tag{13}$$

Then, we prove the following lemma for pessimism in V values.

Lemma 19 (Pessimism for Estimated V Values) *Under the condition of Eq.(12), for any $i \in [n], \pi \in \Pi, x \in \mathcal{S}$, we have*

$$V^{\pi}_{i,h}(x) \ge \widehat{V}^{\pi}_{i,h}(x)\,. \tag{14}$$

Proof For any $i \in [n], \pi \in \Pi, x \in \mathcal{S}, a \in \mathcal{A}$, we have

$$\begin{aligned}
&Q^{\pi}_{i,h}(x,a) - \widehat{Q}^{\pi}_{i,h}(x,a)\\
&\ge r_{i,h}(x,a) + (\mathbb{B}_{i,h}V^{\pi}_{i,h+1})(x,a) - \big(r_{i,h}(s,a) + (\widehat{\mathbb{B}}_{i,h}\widehat{V}^{\pi}_{i,h+1})(x,a) - \Gamma_{i,h}(x,a)\big)\\
&= (\mathbb{B}_{i,h}V^{\pi}_{i,h+1})(x,a) - (\mathbb{B}_{i,h}\widehat{V}^{\pi}_{i,h+1})(x,a) + \Gamma_{i,h}(x,a)\\
&\quad - \big((\widehat{\mathbb{B}}_{i,h}\widehat{V}^{\pi}_{i,h+1})(x,a) - \mathbb{B}_{i,h}\widehat{V}^{\pi}_{i,h+1})(x,a)\big)\\
&\ge (\mathbb{B}_{i,h}V^{\pi}_{i,h+1})(x,a) - (\mathbb{B}_{i,h}\widehat{V}^{\pi}_{i,h+1})(x,a)\\
&= \big(P_{i,h}(V^{\pi}_{i,h+1} - \widehat{V}^{\pi}_{i,h+1})\big)(x,a)\,,
\end{aligned}$$

where the second inequality is because of Eq.(12). And since in the $H+1$ step we have $V^{\pi}_{i,H+1} = \widehat{V}^{\pi}_{i,h+1} = 0$, we can get $Q^{\pi}_{i,H}(x,a) - \widehat{Q}^{\pi}_{i,H}(x,a)$. Then we use induction to prove $Q^{\pi}_{i,h}(x,a) \ge \widehat{Q}^{\pi}_{i,h}(x,a)$ for all h. Given $Q^{\pi}_{i,h+1}(x,a) \ge \widehat{Q}^{\pi}_{i,h+1}(x,a)$, we have

$$\begin{aligned}
Q^{\pi}_{i,h}(x,a) - \widehat{Q}^{\pi}_{i,h}(x,a) &\ge \big(P_{i,h}(V^{\pi}_{i,h+1} - \widehat{V}^{\pi}_{i,h+1})\big)(x,a)\\
&= \mathbb{E}\left[\langle Q^{\pi}_{i,h+1}(s_{h+1},\cdot) - \widehat{Q}^{\pi}_{i,h+1}(s_{h+1},\cdot), \pi_{h+1}(\cdot|s_{h+1})\rangle_{\mathcal{A}} | s_h = x, a_h = a\right]\\
&\ge 0\,.
\end{aligned} \tag{15}$$

Then we have

$$V^{\pi}_{i,h}(x) - \widehat{V}^{\pi}_{i,h}(x) = \langle Q^{\pi}_{i,h}(x,\cdot) - \widehat{Q}^{\pi}_{i,h}(x,\cdot), \pi_h(\cdot\,|\,x)\rangle_{\mathcal{A}} \ge 0\,.$$

■

Then we start our proof.

Proof [Proof of Theorem 5]

First, we decompose the suboptimality gap as follows

$$\text{SubOpt}(\pi^{\text{PERM}})$$

$$
\begin{aligned}
&= \mathbb{E}_{c\sim C} V_{c,1}^{\pi^*}(x_1) - V_{c,1}^{\widehat{\pi}^*}(x_1) \\
&= \mathbb{E}_{c\sim C} V_{c,1}^{\pi^*}(x_1) - \frac{1}{n}\sum_{i=1}^{n} V_{i,1}^{\pi^*}(x_1) + \frac{1}{n}\sum_{i=1}^{n} V_{i,1}^{\pi^{\text{PERM}}}(x_1) - \mathbb{E}_{c\sim C} V_{c,1}^{\pi^{\text{PERM}}}(x_1) \\
&\quad + \frac{1}{n}\sum_{i=1}^{n} \left(V_{i,1}^{\pi^*}(x_1) - V_{i,1}^{\pi^{\text{PERM}}}(x_1)\right). \qquad (16)
\end{aligned}
$$

For the first two terms, we can bound them following the standard generalization techniques (Ye et al. [2023]), *i.e.*, we use the covering argument, Chernoff bound,and union bound.

Define the distance between polices $d(\pi^1, \pi^2) \triangleq \max_{s\in\mathcal{S}, h\in[H]} \|\pi_h^1(\cdot|s) - \pi_h^2(\cdot|s)\|_1$. We construct the ϵ-covering set $\widetilde{\Pi}$ w.r.t. d such that

$$
\forall \pi \in \Pi, \exists \widetilde{\pi} \in \widetilde{\Pi}, s.t. \quad d(\pi, \widetilde{\pi}) \le \epsilon. \qquad (17)
$$

Then we have

$$
\forall i \in [n], \pi \in \Pi, \exists \widetilde{\pi} \in \widetilde{\Pi}, s.t. V_{i,1}^{\pi}(x_1) - V_{i,1}^{\widetilde{\pi}}(x_1) \le H\epsilon. \qquad (18)
$$

By the definition of the covering number, $\left|\widetilde{\Pi}\right| = \mathcal{N}_\epsilon^\Pi$. By Chernoff bound and union bound over the policy set $\widetilde{\Pi}$, we have with prob. at least $1 - \frac{\delta}{3}$, for any $\widetilde{\pi} \in \widetilde{\Pi}$,

$$
\left|\frac{1}{n}\sum_{i=1}^{n} V_{i,1}^{\widetilde{\pi}}(x_1) - \mathbb{E}_{c\sim C} V_{c,1}^{\widetilde{\pi}}(x_1)\right| \le \sqrt{\frac{2\log(6\mathcal{N}_\epsilon^\Pi/\delta)}{n}}. \qquad (19)
$$

By Eq.(18) and Eq.(19), $\forall i \in [n], \pi \in \Pi, \exists \widetilde{\pi} \in \widetilde{\Pi}$ with $\left|\widetilde{\Pi}\right| = \mathcal{N}_\epsilon^\Pi,\ s.t. V_{i,1}^{\pi}(x_1) - V_{i,1}^{\widetilde{\pi}}(x_1) \le H\epsilon$, and with probability at least $1 - \delta/3$, we have

$$
\begin{aligned}
&\left|\frac{1}{n}\sum_{i=1}^{n} V_{i,1}^{\pi}(x_1) - \mathbb{E}_{c\sim C} V_{c,1}^{\pi}(x_1)\right| \\
&\le \left|\frac{1}{n}\sum_{i=1}^{n} V_{i,1}^{\widetilde{\pi}}(s_1) - \mathbb{E}_{c\sim C} V_{c,1}^{\widetilde{\pi}}(x_1)\right| \\
&+ \left|\frac{1}{n}\sum_{i=1}^{n} V_{i,1}^{\pi}(s_1) - \frac{1}{n}\sum_{i=1}^{n} V_{i,1}^{\widetilde{\pi}}(s_1)\right| + \left|\mathbb{E}_{c\sim C} V_{c,1}^{\widetilde{\pi}}(x_1) - \mathbb{E}_{c\sim C} V_{c,1}^{\pi}(x_1)\right| \\
&\le \sqrt{\frac{2\log(6\mathcal{N}_\epsilon^\Pi/\delta)}{n}} + 2H\epsilon\,. \qquad (20)
\end{aligned}
$$

Therefore, we have for the first two terms, w.p. at least $1 - \frac{2}{3}\delta$ we can upper bound them with $4H\epsilon + 2\sqrt{\frac{2\log(6\mathcal{N}_\epsilon^\Pi/\delta)}{n}}$.

Then, what remains is to bound the term $\frac{1}{n}\sum_{i=1}^{n}\left(V_{i,1}^{\pi^*}(x_1) - V_{i,1}^{\pi^{\text{PERM}}}(x_1)\right)$.

First, by similar arguments, we have

$$
\begin{aligned}
V_{i,1}^{\pi^*}(x_1) - V_{i,1}^{\pi^{\text{PERM}}}(x_1) &\le \left(V_{i,1}^{\pi^*}(x_1) - V_{i,1}^{\widetilde{\pi}^{\text{PERM}}}(x_1)\right) + |V_{i,1}^{\widetilde{\pi}^{\text{PERM}}}(x_1) - V_{i,1}^{\pi^{\text{PERM}}}(x_1)| \\
&\le H\epsilon + V_{i,1}^{\pi^*}(x_1) - V_{i,1}^{\widetilde{\pi}^{\text{PERM}}}(x_1)\,, \qquad (21)
\end{aligned}
$$

where $\widetilde{\pi}^{\text{PERM}} \in \widetilde{\Pi}$ such that $|V_{i,1}^{\widetilde{\pi}^{\text{PERM}}}(x_1) - V_{i,1}^{\pi^{\text{PERM}}}(x_1)| \le H\epsilon$.

By the definition of the oracle in Definition.3, the algorithm design of Algo.1 (e.g., we call oracle $\mathbb{O}(\mathcal{D}_h, \widehat{V}_{h+1}, \delta/(3nH\mathcal{N}_{(Hn)^{-1}}^\Pi))$), and use a union bound over H steps, n contexts, and $\mathcal{N}_{(Hn)^{-1}}^\Pi$ policies, we have: with probability at least $1 - \delta/3$, the condition in Eq.(12) holds (with the policy class Π replaced by $\widetilde{\Pi}$ (and $\epsilon = 1/(Hn)$).

Then, we have

$$
\begin{aligned}
&\frac{1}{n}\sum_{i=1}^{n}\left(V_{i,1}^{\pi^*}(x_1)-V_{i,1}^{\widetilde{\pi}^{\text{PERM}}}(x_1)\right)\\
&\leq \frac{1}{n}\sum_{i=1}^{n}\left(V_{i,1}^{\pi^*}(x_1)-\widehat{V}_{i,1}^{\widetilde{\pi}^{\text{PERM}}}(x_1)\right)\\
&= \frac{1}{n}\sum_{i=1}^{n}\left(V_{i,1}^{\pi^*}(x_1)-\widehat{V}_{i,1}^{\pi^{\text{PERM}}}(x_1)\right)+\frac{1}{n}\sum_{i=1}^{n}\left(\widehat{V}_{i,1}^{\pi^{\text{PERM}}}(x_1)-\widehat{V}_{i,1}^{\widetilde{\pi}^{\text{PERM}}}(x_1)\right)\\
&\leq \frac{1}{n}\sum_{i=1}^{n}\left(V_{i,1}^{\pi^*}(x_1)-\widehat{V}_{i,1}^{\pi^{\text{PERM}}}(x_1)\right)+H\cdot\frac{1}{Hn}\\
&\leq \frac{1}{n}\sum_{i=1}^{n}\left(V_{i,1}^{\pi^*}(x_1)-\widehat{V}_{i,1}^{\pi^*}(x_1)\right)+1/n\,,
\end{aligned}
\tag{22}
$$

where the first inequality holds because of the pessimism in Lemma 19, the second inequality holds because $|\widehat{V}_{i,1}^{\widetilde{\pi}^{\text{PERM}}}(x_1)-\widehat{V}_{i,1}^{\pi^{\text{PERM}}}(x_1)|\leq H\epsilon$ with ϵ here specified as $1/(Hn)$, and the last inequality holds because that in the algorithm design of Algo.2 we set $\pi^{\text{PERM}}=\text{argmax}_{\pi\in\Pi}\frac{1}{n}\sum_{i=1}^{n}\widehat{V}_{i,1}^{\pi}(x_1)$.

Then what left is to bound $V_{i,1}^{\pi^*}(x_1)-\widehat{V}_{i,1}^{\pi^*}(x_1)$.

And using Lemma A.1 in Jin et al. [2021], we have

$$
\begin{aligned}
V_{i,1}^{\pi^*}(x_1)-\widehat{V}_{i,1}^{\pi^*}(x_1) &= -\sum_{h=1}^{H}\mathbb{E}_{\widehat{\pi}^*,\mathcal{M}_i}\left[\iota_{i,h}^{\pi^*}(s_h,a_h)\,\middle|\,s_1=x\right]+\sum_{h=1}^{H}\mathbb{E}_{\pi^*,\mathcal{M}_i}\left[\iota_{i,h}^{\pi^*}(s_h,a_h)\,\middle|\,s_1=x\right]\\
&\quad+\sum_{h=1}^{H}\mathbb{E}_{\pi^*,\mathcal{M}_i}\left[\langle\widehat{Q}_{i,h}^{\pi^*}(s_h,\cdot),\pi_h^*(\cdot\,|\,s_h)-\pi_h^*(\cdot\,|\,s_h)\rangle_{\mathcal{A}}\,\middle|\,s_1=x\right]\\
&\leq 2\sum_{h=1}^{H}\mathbb{E}_{\pi^*,\mathcal{M}_i}\left[\Gamma_{i,h}(s_h,a_h)\,\middle|\,s_1=x\right],
\end{aligned}
\tag{23}
$$

where in the last inequality we use Lemma 18.

Finally, with Eq.(16), Eq.(20), Eq.(21), Eq.(22), and Eq.(23), with ϵ set as $\frac{1}{nH}$, we can get w.p. at least $1-\delta$

$$
\begin{aligned}
&\mathbb{E}_{c\sim C}V_{c,1}^{\pi^*}(x_1)-V_{c,1}^{\pi^{\text{PERM}}}(x_1)\\
&\leq \frac{5}{n}+2\sqrt{\frac{2\log(6\mathcal{N}_{(Hn)^{-1}}^{\Pi}/\delta)}{n}}+\frac{2}{n}\sum_{i=1}^{n}\sum_{h=1}^{H}\mathbb{E}_{\pi^*,\mathcal{M}_i}\left[\Gamma_{i,h}(s_h,a_h)|s_1=x_1\right]\\
&\leq 7\sqrt{\frac{2\log(6\mathcal{N}_{(Hn)^{-1}}^{\Pi}/\delta)}{n}}+\frac{2}{n}\sum_{i=1}^{n}\sum_{h=1}^{H}\mathbb{E}_{\pi^*,\mathcal{M}_i}\left[\Gamma_{i,h}(s_h,a_h)|s_1=x_1\right].
\end{aligned}
$$

■

C.2 Proof of Theorem 9

Our proof has two steps. First, we define that

$$
\iota_{i,h}(x,a):=\mathbb{B}_{i,h}V_{i,h+1}(x,a)-Q_{i,h}(x,a) \tag{24}
$$

Then we have the following lemma from Jin et al. [2021]:

Lemma 20 *Define the event $\mathcal{E}$ as*

$$\mathcal{E} = \left\{ \left| (\widehat{\mathbb{B}}\widehat{V}^{\pi_i}_{i,h+1})(x,a) - (\mathbb{B}_{i,h}\widehat{V}^{\pi_i}_{i,h+1})(x,a) \right| \leq \Gamma_{i,h}(x,a) \; \forall (x,a) \in \mathcal{S} \times \mathcal{A}, \forall h \in [H], \forall i \in [n] \right\},$$

Then by selecting the input parameter $\xi = \delta/(Hn)$ in $\mathbb{O}$, we have $\mathbb{P}(\mathcal{E}) \geq 1 - \delta$ and

$$0 \leq \iota_{i,h}(x,a) \leq 2\Gamma_{i,h}(x,a).$$

Proof The proof is the same as [Lemma 5.1, Jin et al. 2021] with the probability assigned as $\delta/(Hn)$ and a union bound over $h \in [H], i \in [n]$. ■

Next lemma shows the difference between the value of the optimal policy π^* and number n of different policies π_i for n MDPs.

Lemma 21 *Let π be an arbitrary policy. Then we have*

$$\begin{aligned} \sum_{i=1}^{n} [V^{\pi}_{i,1}(x_1) - V^{\pi^i}_{i,1}(x_1)] = &\sum_{i=1}^{n}\sum_{h=1}^{H} \mathbb{E}_{i,\pi}[\langle Q_{i,h}(\cdot,\cdot), \pi_h(\cdot|\cdot) - \pi_{i,h}(\cdot|\cdot)\rangle_{\mathcal{A}}] \\ &+ \sum_{i=1}^{n}\sum_{h=1}^{H} (\mathbb{E}_{i,\pi}[\iota_{i,h}(x_h,a_h)] - \mathbb{E}_{i,\pi_i}[\iota_{i,h}(x_h,a_h)]) \end{aligned} \tag{25}$$

Proof The proof is the same as Lemma 3.1 in Jin et al. [2021] except substituting π into the lemma. ■

We also have the following one-step lemma:

Lemma 22 (Lemma 3.3, Cai et al. 2020) *For any distribution $p^*, p \in \Delta(\mathcal{A})$, if $p'(\cdot) \propto p(\cdot)\cdot\exp(\alpha\cdot Q(x,\cdot))$, then*

$$\langle Q(x,\cdot), p^*(\cdot) - p(\cdot)\rangle \leq \alpha H^2/2 + \alpha^{-1} \cdot \Big(\mathit{KL}(p^*(\cdot)\|p(\cdot)) - \mathit{KL}(p^*(\cdot)\|p'(\cdot)) \Big).$$

Given the above lemmas, we begin our proof of Theorem 9.

Proof [Proof of Theorem 9] Combining Lemma 20 and Lemma 21, we have

$$\begin{aligned} &\sum_{i=1}^{n} [V^{\pi^*}_{i,1}(x_1) - V^{\pi^i}_{i,1}(x_1)] \\ &\leq \sum_{i=1}^{n}\sum_{h=1}^{H} \mathbb{E}_{i,\pi^*}[\langle Q_{i,h}, \pi^*_h - \pi_{i,h}\rangle] + 2\sum_{i=1}^{n}\sum_{h=1}^{H} \mathbb{E}_{i,\pi^*}[\Gamma_{i,h}(x_h,a_h)] \\ &\leq \sum_{i=1}^{n}\sum_{h=1}^{H} \alpha H^2/2 + \alpha^{-1}\mathbb{E}_{i,\pi^*}[\mathrm{KL}(\pi^*_h(\cdot|x_h)\|\pi_{i,h}(\cdot|x_h)) - \mathrm{KL}(\pi^*_h(\cdot|x_h)\|\pi_{i+1,h}(\cdot|x_h))] \\ &\quad + 2\sum_{i=1}^{n}\sum_{h=1}^{H} \mathbb{E}_{i,\pi^*}[\Gamma_{i,h}(x_h,a_h)] \\ &\leq \alpha H^3 n/2 + \alpha^{-1} \cdot \sum_{h=1}^{H} \mathbb{E}_{i,\pi^*}[\mathrm{KL}(\pi^*_h(\cdot|x_h)\|\pi_{1,h}(\cdot|x_h))] + 2\sum_{i=1}^{n}\sum_{h=1}^{H} \mathbb{E}_{i,\pi^*}[\Gamma_{i,h}(x_h,a_h)] \\ &\leq \alpha H^3 n/2 + \alpha^{-1} H \log|A| + 2\sum_{i=1}^{n}\sum_{h=1}^{H} \mathbb{E}_{i,\pi^*}[\Gamma_{i,h}(x_h,a_h)], \end{aligned}$$

where the last inequality holds since $\pi_{1,h}$ is the uniform distribution over $\mathcal{A}$. Then, selecting $\alpha = 1/\sqrt{H^2 n}$, we have

$$\sum_{i=1}^{n} [V^{\pi^*}_{i,1}(x_1) - V^{\pi^i}_{i,1}(x_1)] \leq 2\sqrt{n\log|A|H^2} + 2\sum_{i=1}^{n}\sum_{h=1}^{H} \mathbb{E}_{i,\pi^*}[\Gamma_{i,h}(s_h,a_h)],$$

which holds for the random selection of $\mathcal{D}$ with probability at least $1-\delta$. Meanwhile, note that each MDP M_i is drawn i.i.d. from C. Meanwhile, note that π_i only depends on MDP $M_1, ..., M_{i-1}$. Therefore, by the standard online-to-batch conversion, we have

$$\mathbb{P}\bigg(\frac{1}{n}\sum_{i=1}^{n}[V_{i,1}^{\pi^*}(x_1) - V_{i,1}^{\pi_i}(x_1)] + \bigg(\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{c\sim C}V_{c,1}^{\pi_i}(x_1) - \mathbb{E}_{c\sim C}V_{c,1}^{\pi^*}(x_1)\bigg) \leq 2H\sqrt{\frac{2\log 1/\delta}{n}}\bigg) \geq 1-\delta,$$

which suggests that with probability at least $1-2\delta$,

$$\mathbb{E}_{c\sim C}V_{c,1}^{\pi^*}(x_1) - \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}_{c\sim C}V_{c,1}^{\pi_i}(x_1) \leq 2\sqrt{\frac{\log|A|H^2}{n}} + \frac{2}{n}\sum_{i=1}^{n}\sum_{h=1}^{H}\mathbb{E}_{\pi^*}[\Gamma_{i,h}(x_h,a_h)] + 2\sqrt{\frac{2H\log 1/\delta}{n}}.$$

Therefore, by selecting $\pi^{\text{PPPO}} := \text{random}(\pi_1, ..., \pi_n)$ and applying the Markov inequality, setting $\delta = 1/8$, we have our bound holds. ■

D Proof of Theorems in Section 5

D.1 Proof of Theorem 12

By Lemma 5.2 and its proof in Jin et al. [2021], the parameters specified as $\lambda = 1$, $\beta(\delta) = c\cdot dH\sqrt{\log(2dHK/\delta)}$, and applying union bound, we can get: for Algo.4, with probability at least $1-\delta/3$

$$\big|(\widehat{\mathbb{B}}_{i,h}\widehat{V}_{i,h+1}^{\pi})(x,a) - (\mathbb{B}_{i,h}\widehat{V}_{i,h+1}^{\pi})(x,a)\big| \leq \beta\big(\frac{\delta}{3nH\mathcal{N}_{(Hn)^{-1}}^{\Pi}}\big)\big(\phi(x,a)^\top\Lambda_{i,h}^{-1}\phi(x,a)\big)^{1/2},$$
$$\text{for all } i\in[n], \pi\in\widetilde{\Pi}, (x,a)\in\mathcal{S}\times\mathcal{A}, h\in[H], \tag{26}$$

where $\widetilde{\Pi}$ is the $\frac{1}{Hn}$-covering set of the policy space Π w.r.t. distance $\mathrm{d}(\pi^1,\pi^2) = \max_{s\in\mathcal{S},h\in[H]}\|\pi_h^1(\cdot|s) - \pi_h^2(\cdot|s)\|_1$.

Therefore, we can specify the $\Gamma_{i,h}(\cdot,\cdot)$ in Theorem 5 with $\beta\big(\frac{\delta}{3nH\mathcal{N}_{(Hn)^{-1}}^{\Pi}}\big)\big(\phi(x,a)^\top\Lambda_{i,h}^{-1}\phi(x,a)\big)^{1/2}$, and follow the same process as the proof of Theorem 5 to get the result for Algo.2 with subroutine Algo.4.

Similarly, we can get: we can get: for Algo.4, with probability at least $1-1/4$

$$\big|(\widehat{\mathbb{B}}_{i,h}\widehat{V}_{i,h+1})(x,a) - (\mathbb{B}_{i,h}\widehat{V}_{i,h+1})(x,a)\big| \leq \beta\big(\frac{\delta}{4nH}\big)\big(\phi(x,a)^\top\Lambda_{i,h}^{-1}\phi(x,a)\big)^{1/2},$$
$$\text{for all } i\in[n], (x,a)\in\mathcal{S}\times\mathcal{A}, h\in[H]. \tag{27}$$

Therefore, we can specify the $\Gamma_{i,h}(\cdot,\cdot)$ in Theorem 9 with $\beta\big(\frac{\delta}{4nH}\big)\big(\phi(x,a)^\top\Lambda_{i,h}^{-1}\phi(x,a)\big)^{1/2}$ and follow the same process as the proof of Theorem 9 to get the result for Algo.3 with subroutine Algo.4.

D.2 Proof of Corollary 14

By the assumption that $\mathcal{D}_i$ is generated by behavior policy $\bar{\pi}_i$ which well-explores MDP $\mathcal{M}_i$ with constant c_i (where the well-explore is defined in Def.13), the proof of Corollary 4.6 in Jin et al. [2021], and applying a union bound over n contexts, we have that for Algo.2 with subroutine Algo.4 w.p. at least $1-\delta/2$

$$\|\phi(x,a)\|_{\Lambda_{i,h}^{-1}} \leq \sqrt{\frac{2d}{c_iK}} \text{ for all } i\in[n],\ (x,a)\in\mathcal{S}\times\mathcal{A} \text{ and all } h\in[H], \tag{28}$$

and for Algo.2 with subroutine Algo.4 w.p. at least $1-\delta/2$

$$\|\phi(x,a)\|_{\Lambda_{i,h}^{-1}} \leq \sqrt{\frac{2dH}{c_iK}} \text{ for all } i\in[n],\ (x,a)\in\mathcal{S}\times\mathcal{A} \text{ and all } h\in[H], \tag{29}$$

because we use the data splitting technique and we only utilize each trajectory once for one data tuple at some stage h, so we replace K with K/H.

Then, the result follows by plugging the results above into Theorem12.

E Results in Section A

E.1 Proof of Proposition 17

Proof [Proof of Proposition 17] Let $\mathcal{D}' = \{(x^\tau_{c_\tau,h}, a^\tau_{c_\tau,h}, r^\tau_{c_\tau,h})\}_{h=1,\tau=1}^{H,K}$ denote the merged dataset, where each trajectory belongs to a context c_τ. For simplicity, let $\mathcal{D}_c$ denote the collection of trajectories that belong to MDP $\mathcal{M}_c$. Then each trajectory in $\mathcal{D}'$ is generated by the following steps:

- The experimenter randomly samples an environment $c \sim C$.
- The experimenter collect a trajectory from the episodic MDP $\mathcal{M}_c$.

Then for any x', r', τ we have

$$\begin{aligned}
&\mathbb{P}_{\mathcal{D}'}(r^\tau_{c_\tau,h} = r', x^\tau_{c_\tau,h+1} = x' | \{(x^j_{c_j,h}, a^j_{c_j,h})\}_{j=1}^\tau, \{r^j_{c_j,h}, x^j_{c_j,h+1}\}_{j=1}^{\tau-1}) \\
&= \frac{\mathbb{P}_{\mathcal{D}'}(r^\tau_{c_\tau,h} = r', x^\tau_{c_\tau,h+1} = x', \{(x^j_{c_j,h}, a^j_{c_j,h})\}_{j=1}^\tau, \{r^j_{c_j,h}, x^j_{c_j,h+1}\}_{j=1}^{\tau-1})}{\mathbb{P}_{\mathcal{D}'}(\{(x^j_{c_j,h}, a^j_{c_j,h})\}_{j=1}^\tau, \{r^j_{c_j,h}, x^j_{c_j,h+1}\}_{j=1}^{\tau-1})} \\
&= \sum_{c\in C} \mathbb{P}_{\mathcal{D}'}(r^\tau_{c_\tau,h} = r', x^\tau_{c_\tau,h+1} = x' | \{(x^j_{c_j,h}, a^j_{c_j,h})\}_{j=1}^\tau, \{r^j_{c_j,h}, x^j_{c_j,h+1}\}_{j=1}^{\tau-1}, c_\tau = c) q(c), \quad (30)
\end{aligned}$$

where

$$q(c') := \frac{\mathbb{P}_{\mathcal{D}'}(\{(x^j_{c_j,h}, a^j_{c_j,h})\}_{j=1}^\tau, \{r^j_{c_j,h}, x^j_{c_j,h+1}\}_{j=1}^{\tau-1}, c_\tau = c')}{\sum_{c\in C} \mathbb{P}_{\mathcal{D}'}(\{(x^j_{c_j,h}, a^j_{c_j,h})\}_{j=1}^\tau, \{r^j_{c_j,h}, x^j_{c_j,h+1}\}_{j=1}^{\tau-1}, c_\tau = c)}.$$

Next, we further have

$$\begin{aligned}
&(30) \\
&= \sum_{c\in C} \mathbb{P}_c(r_{c,h}(s_h) = r', s_{h+1} = x' | s_h = x^\tau_{c_\tau,h}, a_h = a^\tau_{c_\tau,h}) q(c) \\
&= \sum_{c\in C} \frac{\mathbb{P}_c(r_{c,h}(s_h) = r', s_{h+1} = x' | s_h = x^\tau_{c_\tau,h}, a_h = a^\tau_{c_\tau,h}) \mathbb{P}_{\mathcal{D}'}(s_h = x^\tau_{c_\tau,h}, a_h = a^\tau_{c_\tau,h}, c_\tau = c)}{\sum_{c\in C} \mathbb{P}_{\mathcal{D}'}(s_h = x^\tau_{c_\tau,h}, a_h = a^\tau_{c_\tau,h}, c_\tau = c)} \\
&= \sum_{c\in C} p(c) \cdot \frac{\mathbb{P}_c(r_{c,h}(s_h) = r', s_{h+1} = x' | s_h = x^\tau_{c_\tau,h}, a_h = a^\tau_{c_\tau,h}) \mathbb{P}_c(s_h = x^\tau_{c_\tau,h}, a_h = a^\tau_{c_\tau,h})}{\sum_{c\in C} p(c) \cdot \mathbb{P}_c(s_h = x^\tau_{c_\tau,h}, a_h = a^\tau_{c_\tau,h})} \\
&= \mathbb{E}_{c\sim C} \frac{\mathbb{P}_c(r_{c,h}(s_h) = r', s_{h+1} = x' | s_h = x^\tau_{c_\tau,h}, a_h = a^\tau_{c_\tau,h}) \mu_{c,h}(x^\tau_{c_\tau,h}, a^\tau_{c_\tau,h})}{\mathbb{E}_{c\sim C} \mu_{c,h}(x^\tau_{c_\tau,h}, a^\tau_{c_\tau,h})},
\end{aligned}$$

where the first equality holds since for all trajectories τ satisfying $c_\tau = c$, they are compliant with $\mathcal{M}_c$, the second one holds since all trajectories are independent of each other, the third and fourth ones hold due to the definition of $\mu_{c,h}(\cdot,\cdot)$. ■

E.2 PEVI algorithm

Algorithm 6 [Jin et al., 2021] Pessimistic Value Iteration (PEVI)

Require: Dataset $\mathcal{D} = \{(x^\tau_{c_\tau,h}, a^\tau_{c_\tau,h}, r^\tau_{c_\tau,h})_{h=1}^H\}_{\tau=1}^K$, confidence probability $\delta \in (0,1)$.

1: Initialization: Set $\widehat{V}_{H+1}(\cdot) \leftarrow 0$.
2: **for** step $h = H, H-1, \ldots, 1$ **do**
3: Set $\Lambda_h \leftarrow \sum_{\tau=1}^K \phi(x_h^\tau, a_h^\tau)\phi(x_h^\tau, a_h^\tau)^\top + \lambda \cdot I$.
4: Set $\widehat{w}_h \leftarrow \Lambda_h^{-1}(\sum_{\tau=1}^K \phi(x_h^\tau, a_h^\tau) \cdot (r_h^\tau + \widehat{V}_{h+1}(x_{h+1}^\tau)))$.
5: Set $\Gamma_h(\cdot,\cdot) \leftarrow \beta(\delta) \cdot (\phi(\cdot,\cdot)^\top \Lambda_h^{-1} \phi(\cdot,\cdot))^{1/2}$.
6: Set $\widehat{Q}_h(\cdot,\cdot) \leftarrow \min\{\phi(\cdot,\cdot)^\top \widehat{w}_h - \Gamma_h(\cdot,\cdot), H-h+1\}^+$.
7: Set $\widehat{\pi}_h(\cdot\,|\,\cdot) \leftarrow \mathrm{argmax}_{\pi_h} \langle \widehat{Q}_h(\cdot,\cdot), \pi_h(\cdot\,|\,\cdot)\rangle_{\mathcal{A}}$.
8: Set $\widehat{V}_h(\cdot) \leftarrow \langle \widehat{Q}_h(\cdot,\cdot), \widehat{\pi}_h(\cdot\,|\,\cdot)\rangle_{\mathcal{A}}$.
9: **end for**

Ensure: $\pi^{\text{PEVI}} = \{\widehat{\pi}_h\}_{h=1}^H$.

We analyze the suboptimality gap of the Pessimistic Value Iteration (PEVI) (Jin et al. [2021]) in the contextual linear MDP setting without context information to demonstrate that by finding the optimal policy for $\bar{\mathcal{M}}$ is not enough to find the policy that performs well on MDPs with context information.

Pessimistic Value Iteration (PEVI). Let $\overline{\pi}^*$ be the optimal policy w.r.t. the average MDP $\bar{\mathcal{M}}$. We analyze the performance of the Pessimistic Value Iteration (PEVI) [Jin et al., 2021] under the unknown context information setting. The details of PEVI is in Algo.6.

Suppose that $\bar{\mathcal{D}}$ consists of K number of trajectories generated i.i.d. following by a fixed behavior policy $\bar{\pi}$. Then the following theorem shows the suboptimality gap for Algo.6 does not converge to 0 even when the data size grows to infinity.

Theorem 23 *Assume that $\bar{\pi}$ In Algo.4, we set*

$$\lambda = 1, \quad \beta(\delta) = c' \cdot dH\sqrt{\log(4dHK/\delta)}\,, \tag{31}$$

where $c' > 0$ is a positive constant. Suppose we have $K \geq \widetilde{c} \cdot d\log(4dH/\xi)$, where $\widetilde{c} > 0$ is a sufficiently large positive constant that depends on c. Then we have: w.p. at least $1-\delta$, for the output policy π^{PEVI} of Algo.6,

$$\sup_\pi V^\pi_{\bar{\mathcal{M}},1} - V^{\pi^{PEVI}}_{\bar{\mathcal{M}},1} \leq c'' \cdot d^{3/2}H^2K^{-1/2}\sqrt{\log(4dHK/\delta)}, \tag{32}$$

and the suboptimality gap satisfies

$$\text{SubOpt}(\pi^{\text{PEVI}}) \leq c'' \cdot d^{3/2}H^2K^{-1/2}\sqrt{\log(4dHK/\delta)} + 2\sup_\pi |V^\pi_{\bar{\mathcal{M}},1}(x_1) - \mathbb{E}_{c\sim C}V^\pi_{c,1}(x_1)|\,, \tag{33}$$

where $c'' > 0$ is a positive constant that only depends on c and c'.

Proof [Proof of Theorem 23] First, we define the value function on the average MDP $\bar{\mathcal{M}}$ as follows.

$$\overline{V}^\pi_h(x) = \mathbb{E}_{\pi,\bar{\mathcal{M}}}\Big[\sum_{i=h}^H r_i(s_i,a_i)\,\Big|\,s_h = x\Big]\,. \tag{34}$$

We then decompose the suboptimality gap as follows.

$$\begin{aligned}
&\text{SubOpt}(\pi^{\text{PEVI}}) \\
&= \mathbb{E}_{c\sim C}\big[V^{\pi^*}_{c,1}(x_1)\big] - \mathbb{E}_{c\sim C}\big[V^{\pi^{\text{PEVI}}}_{c,1}(x_1)\big] \\
&= \overline{V}^{\overline{\pi}^*}_1(x_1) - \overline{V}^{\pi^{\text{PEVI}}}_1(x_1) + \big(\mathbb{E}_{c\sim C}\big[V^{\pi^*}_{c,1}(x_1)\big] - \overline{V}^{\overline{\pi}^*}_1(x_1)\big) + \big(\overline{V}^{\pi^{\text{PEVI}}}_1(x_1) - \mathbb{E}_{c\sim C}\big[V^{\pi^{\text{PEVI}}}_{c,1}(x_1)\big]\big)
\end{aligned}$$

$$\leq \overline{V}_1^{\bar{\pi}^*}(x_1) - \overline{V}_1^{\pi^{\text{PEVI}}}(x_1) + 2\sup_{\pi} |V^{\pi}_{\bar{\mathcal{M}},1}(x_1) - \mathbb{E}_{c\sim C} V^{\pi}_{c,1}(x_1)| \,. \tag{35}$$

Then, applying Corollary 4.6 in Jin et al. [2021], we can get that w.p. at least $1-\delta$

$$\overline{V}_1^{\bar{\pi}^*}(x_1) - \overline{V}_1^{\pi^{\text{PEVI}}}(x_1) \leq c'' \cdot d^{3/2} H^2 K^{-1/2} \sqrt{\log(4dHK/\delta)} \,, \tag{36}$$

which, together with Eq.(35) completes the proof.

■

Theorem 23 shows that by adapting the standard pessimistic offline RL algorithm over the offline dataset without context information, the learned policy π^{PEVI} converges to the optimal policy $\bar{\pi}^*$ over the average MDP $\bar{\mathcal{M}}$.