Uncertainty-Aware Reward-Free Exploration with General Function Approximation

Junkai Zhang [*1] **Weitong Zhang** [*1] **Dongruo Zhou** [2] **Quanquan Gu** [1]

Abstract

Mastering multiple tasks through exploration and learning in an environment poses a significant challenge in reinforcement learning (RL). Unsupervised RL has been introduced to address this challenge by training policies with intrinsic rewards rather than extrinsic rewards. However, current intrinsic reward designs and unsupervised RL algorithms often overlook the heterogeneous nature of collected samples, thereby diminishing their sample efficiency. To overcome this limitation, in this paper, we propose a reward-free RL algorithm called `GFA-RFE`. The key idea behind our algorithm is an *uncertainty-aware intrinsic reward* for exploring the environment and an *uncertainty-weighted* learning process to handle heterogeneous uncertainty in different samples. Theoretically, we show that in order to find an ϵ-optimal policy, `GFA-RFE` needs to collect $\widetilde{O}(H^2 \log N_{\mathcal{F}}(\epsilon) \dim(\mathcal{F})/\epsilon^2)$ number of episodes, where $\mathcal{F}$ is the value function class with covering number $N_{\mathcal{F}}(\epsilon)$ and generalized eluder dimension $\dim(\mathcal{F})$. Such a result outperforms all existing reward-free RL algorithms. We further implement and evaluate `GFA-RFE` across various domains and tasks in the DeepMind Control Suite. Experiment results show that `GFA-RFE` outperforms or is comparable to the performance of state-of-the-art unsupervised RL algorithms.

1 Introduction

Deep reinforcement learning (RL) has been the source of many breakthroughs in games (e.g., Atari game (Mnih et al., 2013) and Go game (Silver et al., 2016)) and robotic control (Levine et al., 2016) over the last ten years. A key component of RL is exploration, which requires the agent to explore different states and actions before finding a near-optimal policy. Traditional exploration strategy involves iteratively executing a policy guided by a specific reward function, limiting the trained agent to solving only the single task for which it was trained. Designing an efficient exploration strategy agnostic to reward functions is crucial, as it prevents the agent from repeated learning under different reward functions, thereby avoiding inefficiency and potential intractability in sample complexity.

To achieve this goal, Jin et al. (2020a) introduced a two-phase RL framework known as "reward-free exploration" (RFE) for the basic tabular MDP setting. In this framework, the agent only interacts with the environment in the first phase without reward. Upon receiving the specific reward in the second phase, the algorithm returns a near-optimal policy without further interactions. The overall framework is displayed in Figure 1. A series of subsequent works extended the idea to more complex settings, such as linear MDPs (Wang et al., 2020; Zanette et al., 2020; Wagenmaker et al., 2022; Hu et al., 2022) and linear mixture MDPs (Zhang et al., 2021b; Chen et al., 2021; Zhang et al., 2023). RFE diverges from classical RL approaches by not relying on a specific reward function for exploration. Instead, RFE utilizes an "intrinsic reward", a.k.a., pseudo-reward function, defined based on all previously explored samples. This encourages the agent to venture into unexplored states and actions. In particular, in the realm of deep RL where no structural assumptions are made, recent studies (Pathak et al., 2017; Burda et al., 2018b; Eysenbach et al., 2018; Lee et al., 2019; Pathak et al., 2019; Liu & Abbeel, 2021a;b) have developed RFE (a.k.a., unsupervised RL) algorithms by employing various intrinsic reward functions, demonstrating promising performance in finding the near-optimal policy.

Despite the success of intrinsic reward functions in facilitating RFE, the design of these functions in prior studies could be further optimized. For example, Kong et al. (2021) defined an intrinsic reward based on the maximum difference between function pairs that show similarity in past data. This approach essentially treats each collected sample equally. It is a well-established principle in RL that in order to achieve optimal sample efficiency, different samples

[*]Equal contribution [1]Department of Computer Science, University of California, Los Angeles, California, USA [2]Department of Computer Science, Indiana University Bloomington, Indiana, USA. Correspondence to: Quanquan Gu <qgu@cs.ucla.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

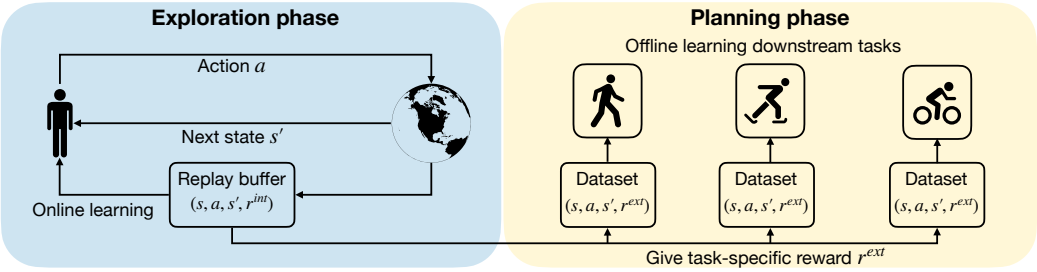


Figure 1. The overall framework of reward-free exploration.

should be treated distinctively based on their importance. Notably, Zhang et al. (2023) utilized variance-dependent weights to address the heteroscedasticity observed in samples, thereby achieving optimal sample complexity in linear mixture MDPs. However, this approach calculates its intrinsic reward by nested iterative optimization, which hampers computational efficiency and practical applicability. Therefore, for RFE or more generally unsupervised RL, we are faced with the following question:

Is it possible to craft an intrinsic reward function that excels both theoretically and empirically?

We answer the above question affirmatively by proposing a variance-adaptive intrinsic reward for RFE. Theoretically, we show that our method enjoys a finite sample complexity in finding the near-optimal policy for any given reward, and our theoretical guarantee is tighter than existing methods. Empirically, we show that by incorporating variance information, a series of existing reward-free RL baselines can be further improved in terms of sample efficiency. The main contributions of our work can be summarized as follows.

- We propose a new algorithm `GFA-RFE` under the RFE framework. The key innovation of `GFA-RFE` is a new intrinsic reward, which depends on an uncertainty estimation of each past state-action pair that appeared during the exploration phase. Our designed intrinsic reward relies more on observed samples with lower uncertainty, and it encourages the agent to explore states and actions with larger uncertainty. Intuitively speaking, such a strategy ensures the agent to find samples that are universally suitable for all reward functions that it may encounter during the planning phase, without knowing the extrinsic rewards in hindsight.
- Theoretically, we prove that during the planning phase, given any reward function r, `GFA-RFE` achieves an $\widetilde{O}(H^2 \log N_{\mathcal{F}}(\epsilon) \dim(\mathcal{F})/\epsilon^2)$ sample complexity to find the ϵ-optimal policy w.r.t. the reward r, where $\mathcal{F}$ is the value function class with covering number $N_{\mathcal{F}}(\epsilon)$ and generalized eluder dimension $\dim(\mathcal{F})$. Our sample complexity outperforms the existing sample complexity result achieved by Kong et al. (2021) (see Table 1), which verifies our claim that an adaptive intrinsic reward improves exploration efficiency.
- We also show that our variance-adaptive intrinsic reward can efficiently explore the environment in practice through extensive experiments on the DeepMind Control Suite (Tassa et al., 2018). Our theory-guided algorithm `GFA-RFE` exhibits compatible or superior performance compared with the state-of-the-art unsupervised exploration methods. This promising result demonstrates the huge potential of incorporating the theories into practice to solve real-world problems.

Notation We denote by $[n]$ the set $\{1, \cdots, n\}$. For two positive sequences $\{a_n\}$ and $\{b_n\}$ with $n = 1, 2, \cdots$, we write $a_n = O(b_n)$ if there exists an absolute constant $C > 0$ such that $a_n \leq C b_n$ holds for all $n \geq 1$, write $a_n = \Omega(b_n)$ if there exists an absolute constant $C > 0$ such that $a_n \geq C b_n$ holds for all $n \geq 1$, and write $a_n = o(b_n)$ if $a_n/b_n \to 0$ as $n \to \infty$. We use $\widetilde{O}(\cdot)$ and $\widetilde{\Omega}(\cdot)$ to further hide the polylogarithmic factors.

2 Related Work

Reinforcement learning with general function approximation. RL with general function approximation has been widely studied in recent years, due to its ability to describe a wide range of existing RL algorithms. To explore the theoretical limits of RL and understand the practical DRL algorithms, various statistical complexity measurements for general function approximation have been proposed and developed. For instance, Bellman rank (Jiang et al., 2017), Witness rank (Sun et al., 2019), eluder dimension (Russo & Van Roy, 2013), Bellman eluder dimension (Jin et al., 2021), Decision-Estimation Coefficient (DEC) (Foster et al., 2021), Admissible Bellman Characterization (Chen et al., 2022c), generalized eluder dimension (Agarwal et al., 2022), etc. Among different statistical complexity measurements, Foster et al. (2021) showed a DEC-based lower bound of regret which holds for any function class. Specifically, our algorithm falls into the category of generalized eluder dimension function class, which includes linear MDPs (Jin et al., 2020b) as its special realization.

Reward-free exploration. Unlike standard RL settings where the agent interacts with the environment with reward signals, *reward-free exploration* (Jin et al., 2020a) in RL in-

Table 1. Comparison of episodic reward-free RL algorithms in different settings. Column **Comp. Eff.** (Computational Efficiency) indicates if the algorithm can be efficiently implemented, or can be segregated by some efficient algorithm. Column **Time Homo.** (Time Homogeneity) indicates if the setting is time homogeneous ✓ or time inhomogeneous ×. Time inhomogeneous settings usually yield an additional H in sample complexity in learning different stages $h \in [H]$. The sample complexity is evaluated under the reward scale $r_h(s_h, a_h) \in [0, 1]$. The results with bounded total reward assumption ($\sum_{h=1}^{H} r_h(s_h, a_h) \leq 1$) are translated by inserting an additional H^2 dependency on the reward scale and marked with (*). Dimension d in general function approximations inherits their original definitions in the paper, and it usually corresponds to the dimension of linear function when reduced to linear function approximations. The row with a light cyan background indicates our results.

Setting	Algorithm	Comp. Eff.	Time Homo.	Sample Complexity
Linear MDP	Wang et al. (2020)	✓	×	$\widetilde{O}\left(H^6 d^3 \epsilon^{-2}\right)$
	FRANCIS (Zanette et al., 2020)	✓	×	$\widetilde{O}\left(H^5 d^3 \epsilon^{-2}\right)$
	RFLIN (Wagenmaker et al., 2022)	✓	×	$\widetilde{O}\left(H^5 d^2 \epsilon^{-2}\right)$
	LSVI-RFE Hu et al. (2022)	✓	×	$\widetilde{\mathcal{O}}(H^4 d^2 \epsilon^{-2})$
Linear Mixture MDP	UCRL-RFE+ (Zhang et al., 2021b)	✓	✓	$\widetilde{O}\left(H^4 d(H+d)\epsilon^{-2}\right)$
	Chen et al. (2021)	×	×	$\widetilde{O}\left(H^3 d(H+d)\epsilon^{-2}\right)$
	HF-UCRL-RFE++ (Zhang et al., 2023)	×	✓	$\widetilde{\mathcal{O}}(H^2 d^2 \epsilon^{-2})^*$
General Function Approximation	Kong et al. (2021)	✓	×	$\widetilde{\mathcal{O}}(H^6 d^4 \epsilon^{-2})$
	Reward-Free E2D (Chen et al., 2022a)	×	×	$\widetilde{\mathcal{O}}(d \log \lvert\mathcal{P}\rvert \epsilon^{-2})$
	RFOlive (Chen et al., 2022b)	×	×	$\widetilde{\mathcal{O}}(\mathrm{poly}(H) d_{\mathrm{BE}}^2 \log(\lvert\mathcal{F}\rvert\lvert\mathcal{R}\rvert)\epsilon^{-2})$
	`GFA-RFE` **(Ours)**	✓	×	$\widetilde{\mathcal{O}}(H^4 d_{K,\delta}^2 \epsilon^{-2})^*$
Linear (Mixture) MDP	Lower bound (Hu et al., 2022)	N/A	×	$\widetilde{\Omega}(H^3 d^2 \epsilon^{-2})$

troduced a two-phase paradigm. In this approach, the agent initially explores the environment without any reward signals. Then, upon receiving the reward functions, it outputs a policy that maximizes the cumulative reward, without any further interaction with the environment. Jin et al. (2020a) first achieved $\widetilde{O}(H^5 S^2 A/\epsilon^2)$ sample complexity in tabular MDPs by executing exploratory policy visiting states with probability proportional to its maximum visitation probability under any possible policy. Subsequent works (Kaufmann et al., 2021; Ménard et al., 2021) proposed algorithms RF-UCRL and RF-Express to gradually improve the result to $\widetilde{O}\left(H^3 S^2 A \epsilon^{-2}\right)$. The optimal sample complexity bound $\widetilde{O}(H^2 S^2 A \epsilon^{-2})$ was achieved by algorithm SSTP proposed in Zhang et al. (2020), which matched the lower bound provided in Jin et al. (2020a) up to logarithmic factors. Recent years have witnessed a trend of reward-free exploration in RL with function approximations, while most of these works are considering linear function approximation: in the linear MDP setting, Wang et al. (2020) propose an exploration-driven reward function and the minimax optimal bound was achieved by Hu et al. (2022) by introducing the weighted regression in the algorithm. In linear mixture MDPs, Zhang et al. (2021c) proposed the 'pseudo reward' to encourage exploration, Chen et al. (2021); Wagenmaker et al. (2022) improved the sample complexity by introducing a more complicated, recursively defined pseudo reward. The minimax optimal sample complexity, $\widetilde{\mathcal{O}}(d^2/\epsilon^2)$ was achieved by Zhang et al. (2023) in the horizon-free setting. Moving forward, in the general function approximation setting, Kong et al. (2021) used 'online sensitivity score' to estimate the information gain thus providing a $\widetilde{\mathcal{O}}(d^4 H^6 \epsilon^{-2})$ sample complexity where d is the dimension of contexts when reduced to linear function approximations. Yet another line of works (Chen et al., 2022a;b) aimed to follow the Decision-Estimation Coefficient (DEC, Foster et al. 2021) and provided a unified framework for reward-free exploration with general function approximations, achieving a $\widetilde{\mathcal{O}}(\mathrm{ploy}(H) d^2 \epsilon^{-2})$, nevertheless, all existing works with general function approximations leave a huge gap between their proposed upper bound and lower bound, even when reduced to linear settings. We record existing results in Table 1.

Unsupervised reinforcement learning. Witnessing recent advancements in unsupervised CV and NLP tasks, unsupervised reinforcement learning has emerged as a new paradigm trying to learn the environment without supervision or reward signals. As suggested in Laskin et al. (2021), these works are mainly separated into two lines: unsupervised representation learning in RL and unsupervised behavioral learning.

Unsupervised representation learning in RL mainly addresses issues on how to learn good representations for different states s, which can facilitate efficient learning of a policy $\pi(a|s)$. From the theoretical side, a list of works have identified how to select or learn good representations for various RL tasks with linear function approximations, by using MLE (Uehara et al., 2021), contrastive learning (Qiu et al., 2022) or model selection (Papini et al., 2021; Zhang et al., 2021a). From the empirical side, various methods in unsupervised learning or self-supervised learning are applied to RL tasks, including contrastive learning (Laskin et al., 2020; Stooke et al., 2021; Yarats et al., 2021a), autoencoders (Yarats et al., 2021b) and world models (Hafner et al., 2019a;b).

Unsupervised behavioral learning in RL aims to eliminate this reward signal during exploration. Therefore, the agent can be adapted to different tasks in the downstream finetuning. To replace the 'extrinsic' reward signals, these methods usually leverage different 'intrinsic rewards' during exploration. Many recent algorithms have been proposed to learn from different types of intrinsic reward, which is based on the prediction, information gain or entropy. In particular, Pathak et al. (2017); Burda et al. (2018a); Pathak et al. (2019) are referred to as "knowledge based" intrinsic reward (Laskin et al., 2021) and they all maintain a neural network $g(s_t, a_t)$ to predict the next state s_{t+1} from the current state and actions. Among these three methods, Pathak et al. (2017) and Burda et al. (2018a) are using the prediction error $|s_{t+1} - g(s_t, a_t)|$ as the intrinsic reward, Pathak et al. (2019) is using the variance of N ensemble neural networks (i.e., Var $g_i(s_t, a_t)$) as the intrinsic reward. On the other hand, Lee et al. (2019); Eysenbach et al. (2018); Liu & Abbeel (2021a) are trying to maximize the mutual information to complete the exploration of the agent, thus they are referred to as the "complete-based" algorithms. In addition, Liu & Abbeel (2021b) is trying to maximize the entropy of the collected observations via a kNN method, which is referred to as the "data-based" algorithm. URLB (Laskin et al., 2021) provided a unified framework providing benchmarks for all these intrinsic rewards.

3 Problem Setup

3.1 Time-Inhomogeneous Episodic MDPs

We model the sequential decision making problem via time-inhomogeneous episodic Markov decision processes (MDPs), which can be denoted as tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, \mathbb{P} = \{\mathbb{P}_h\}_{h=1}^H, r = \{r_h\}_{h=1}^H)$ by convention. Here, $\mathcal{S}$ and $\mathcal{A}$ are state and action spaces, H is the length of each episode, $\mathbb{P}_h : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition probability function at stage h for state s to transit to state s' after executing action a, and $r_h : \mathcal{S} \times \mathcal{A} \to [0, 1]$ is the deterministic reward function at stage h. For any policy $\pi = \{\pi_h\}_{h=1}^H$, reward $r = \{r_h\}_{h=1}^H$, and stage $h \in [H]$, the value function $V_h^\pi(s; r)$ and the state-action value function $Q_h^\pi(s, a; r)$ is defined as:

$$Q_h^\pi(s,a;r) = \mathbb{E}\Bigg[\sum_{h'=h}^{H} r_{h'}\big(s_{h'}, a_{h'}\big)\Bigg| s_h = s, a_h = a, \\ s_{h'+1} \sim \mathbb{P}_{h'}(\cdot|s_{h'}, a_{h'}), a_{h'+1} = \pi(s_{h'+1})\Bigg],$$

$$V_h^\pi(s;r) = Q_h^\pi(s, \pi_h(s); r).$$

Furthermore, the optimal value function $V_h^*(s; r)$ is defines as $\max_\pi V_h^\pi(s; r)$, and the optimal action-value function $Q_h^*(s, a; r)$ is defined as $\max_\pi Q_h^\pi(s, a; r)$. For simplicity, we utilize the following bounded total reward assumption:

Assumption 3.1. The total reward for every possible trajectory is assumed to be within the interval of $(0, 1)$.

Up to rescaling, Assumption 3.1 is more general than the standard reward scale assumption where $r_h \in [0, 1]$ for all $h \in [H]$. Assumption 3.1 also ensures that the value function $V_h^\pi(s)$ and action-value function $Q_h^\pi(s, a; r)$ belong to the interval $[0, 1]$.

For any function $V : \mathcal{S} \to \mathbb{R}$ and stage $h \in [H]$, the first-order Bellman operator $\mathcal{T}_h$ is defined as:

$$\mathcal{T}_h V(s,a;r) = \mathbb{E}_{s'\sim\mathbb{P}(\cdot|s,a)}\Big[r_h(s,a) + V(s';r)\Big].$$

For simplicity, we further define the shorthand:

$$[\mathbb{P}_h V](s,a;r) = \mathbb{E}_{s'\sim\mathbb{P}_h(\cdot|s,a)} V(s';r),$$
$$[\mathbb{V}_h V](s,a;r) = [\mathbb{P}_h V^2](s,a;r) - [\mathbb{P}_h V]^2(s,a;r).$$

Throughout the paper, if the reward r is clear in the context, we omit the notation r in Q and V for simplicity.

3.2 Reward-free Exploration

Reward-free RL. In reward-free RL, the real reward function is accessible only after the agent finishes the interactions with the environment. Specifically, the algorithm can be separated into two phases: (i) *Exploration phase*: the algorithm can't access the reward function but collects K episodes of samples by interacting with the environment. (ii) *Planning phase*: The algorithm is given reward function $\{r_h\}_{h=1}^H$ and is expected to find the optimal policy without interaction with the environment.

To deal with the randomness in learning processes and evaluate the efficiency of algorithms, we adopt the commonly used (ϵ, δ)-learnability concept, which is formulated in Definition 3.2.

Definition 3.2. ((ϵ, δ)-learnability). Given an MDP transition kernel set $\mathcal{P}$, reward function set $\mathcal{R}$ and a initial state distribution μ, we say a reward-free algorithm can (ϵ, δ)-learn the problem $(\mathcal{P}, \mathcal{R})$ with sample complexity $K(\epsilon, \delta)$, if for any transition kernel $P \in \mathcal{P}$, after receiving $K(\epsilon, \delta)$ episodes in the exploration phase, for any reward function $r \in \mathcal{R}$, the algorithm returns a policy π in planning phase, such that with probability at least $1 - \delta$, $V_1^*(s_1; r) - V_1^\pi(s_1; r) \le \epsilon$.

3.3 General Function Approximation

In this work, we focus on the model-free value-based RL methods, which require us to use a predefined function class to estimate the optimal value function $Q_h^*(s,a;r)$ for any reward r. We use $\mathcal{F} := \{\mathcal{F}_h\}_{h=1}^H$ to denote the function class we will use during all H stages. To build the statistical complexity of using $\mathcal{F}$ to learn $Q_h^*(s,a;r)$, we require several assumptions and definitions that characterize the cardinality of the function class.

Assumption 3.3 (Completeness, Zhao et al. (2023))**.** Given $\mathcal{F} := \{\mathcal{F}_h\}_{h=1}^H$ which is composed of bounded functions $f_h : \mathcal{S} \times \mathcal{A} \to [0, L]$. We assume that for any h and function $V : \mathcal{S} \to [0,1]$ and $r : \mathcal{S} \times \mathcal{A} \to [0,1]$, there exist $f_1, f_2 \in \mathcal{F}_h$ such that for any $(s,a) \in \mathcal{S} \times \mathcal{A}$,

$$f_1(s,a) = \mathbb{E}_{s' \sim \mathbb{P}_h(\cdot|s,a)}\big[r(s,a) + V(s')\big],$$
$$f_2(s,a) = \mathbb{E}_{s' \sim \mathbb{P}_h(\cdot|s,a)}\Big[\big(r(s,a) + V(s')\big)^2\Big].$$

We assume that $L = O(1)$ throughout the paper.

Definition 3.4 (Generalized eluder dimension, Agarwal et al. 2022)**.** Let $\lambda \geq 0$ and $h \in [H]$, a sequence of state-action pairs $Z_h = \{z_{i,h} = (s_h^i, a_h^i)\}_{i \in [K]}$ and a sequence of positive numbers $\boldsymbol{\sigma}_h = \{\sigma_{i,h}\}_{i \in [K]}$. The generalized eluder dimension of a function class $\mathcal{F}_h : \mathcal{S} \times \mathcal{A} \to [0, L]$ with respect to λ is defined by $\dim_{\alpha,K}(\mathcal{F}_h) := \sup_{Z_h, \boldsymbol{\sigma}_h : |Z_h| = K, \boldsymbol{\sigma}_h \geq \alpha} \dim(\mathcal{F}_h, Z_h, \boldsymbol{\sigma}_h)$

$$\dim(\mathcal{F}_h, Z_h, \boldsymbol{\sigma}_h) := \sum_{i=1}^K \min\left(1, \frac{1}{\sigma_i^2} D^2_{\mathcal{F}_h}(z_{i,h}; z_{[i-1],h}, \sigma_{[i-1],h})\right),$$

$$D^2_{\mathcal{F}_h}(z; z_{[i-1],h}, \sigma_{[i-1],h}) := \sup_{f_1, f_2 \in \mathcal{F}_h} \frac{(f_1(z) - f_2(z))^2}{\sum_{s \in [i-1]} \frac{1}{\sigma_{s,h}^2}(f_1(z_{s,h}) - f_2(z_{s,h}))^2 + \lambda}.$$

We write $\dim_{\alpha,K}(\mathcal{F}) := H^{-1} \cdot \sum_{h \in [H]} \dim_{\alpha,K}(\mathcal{F}_h)$ for short when $\mathcal{F}$ is a collection of function classes $\mathcal{F} = \{\mathcal{F}_h\}_{h=1}^H$ in the context.

Remark 3.5. Kong et al. (2021) introduced a similar definition called "sensitivity". In particular, it is defined by

$$\text{sensitivity}_{\mathcal{Z},\mathcal{F}}(z) := \sup_{f_1, f_2 \in \mathcal{F}} \frac{(f_1(z) - f_2(z))^2}{\min\{\sum_{(s,a) \in \mathcal{Z}} (f_1(s,a) - f_2(s,a))^2, \lambda\}},$$

where λ is defined by $T(H+1)^2$ for the RL task with $r_h(s,a) \in [0,1]$ [1]. The major difference between the generalized eluder dimension and sensitivity is that the generalized eluder dimension incorporates the variance σ_s^2 into the historical observation $\mathcal{Z}$ to craft the heterogeneous variance in $\mathcal{Z}$.

[1]We ignore the clipping process making $\text{sensitivity}_{\mathcal{Z},\mathcal{F}}(z) \leftarrow \min\{\text{sensitivity}_{\mathcal{Z},\mathcal{F}}(z)\}$ for the clarity of demonstration

Since $D^2_{\mathcal{F}_h}$ in Definition 3.4 is not computationally efficient in some circumstances, we approximate it via an oracle $\overline{D}^2_{\mathcal{F}_h}$, which is formally defined in Definition 3.6.

Definition 3.6 (Bonus oracle $\overline{D}^2_{\mathcal{F}_h}$)**.** The bonus oracle returns a computable function $\overline{D}^2_{\mathcal{F}_h}(z; z_{[t],h}, \sigma_{[t],h})$, which computes the estimated uncertainty of a state-action pair $z = (s,a) \in \mathcal{S} \times \mathcal{A}$ with respect to historical data $z_{[t],h}$ and corresponding weights $\sigma_{[t],h}$. It satisfies

$$D_{\mathcal{F}_h}(z; z_{[t],h}, \sigma_{[t],h}) \leq \overline{D}_{\mathcal{F}_h}(z; z_{[t],h}, \sigma_{[t],h}) \leq C \cdot D_{\mathcal{F}_h}(z; z_{[t],h}, \sigma_{[t],h}),$$

where C is a fixed constant.

The covering numbers of the value function class and the bonus function class are introduced in the following definition.

Definition 3.7 (Covering numbers of function classes)**.** For any $\epsilon > 0$, we define the following covering numbers of involved function classes:

1. For each $h \in [H]$, there exists an ϵ-cover $\mathcal{C}(\mathcal{F}_h, \epsilon) \subseteq \mathcal{F}_h$ with size $|\mathcal{C}(\mathcal{F}_h, \epsilon)| \leq N_{\mathcal{F}_h}(\epsilon)$, such that for any $f \in \mathcal{F}_h$, there exists $f' \in \mathcal{C}(\mathcal{F}_h, \epsilon)$ satisfying $\|f - f'\|_\infty \leq \epsilon$. For any $\epsilon > 0$, we define the uniform covering number of $\mathcal{F}$ with respect to ϵ as $N_{\mathcal{F}}(\epsilon) := \max_{h \in [H]} N_{\mathcal{F}_h}(\epsilon)$.
2. There exists a bonus function class $\mathcal{B} = \{B : \mathcal{S} \times \mathcal{A} \to \mathbb{R}\}$ such that for any $t \geq 0$, $z_{[t]} \in (\mathcal{S} \times \mathcal{A})^t$, $\sigma_{[t]} \in \mathbb{R}^t$, $h \in [H]$, the bonus function $\overline{D}_{\mathcal{F}}(\cdot; z_{[t]}, \sigma_{[t]})$ returned by the bonus oracle in Definition 3.6 belongs to $\mathcal{B}$.
3. For the bonus function class $\mathcal{B}$, there exists an ϵ-cover $\mathcal{C}(\mathcal{B}, \epsilon) \subseteq \mathcal{B}$ with size $|\mathcal{C}(\mathcal{B}, \epsilon)| \leq N_{\mathcal{B}}(\epsilon)$, such that for any $b \in \mathcal{B}$, there exists $b' \in \mathcal{C}(\mathcal{B}, \epsilon)$, such that $\|b - b'\|_\infty \leq \epsilon$.
4. The optimistic function class at stage $h \in [H]$ is:

$$\mathcal{V}_h = \Big\{V(\cdot) = \max_{a \in \mathcal{A}} \min\Big(1, f(\cdot, a) + \beta \cdot b(\cdot, a)\Big) \Big| f \in \mathcal{F}_h, b \in \mathcal{B}\Big\}.$$

There exists an ϵ-cover $\mathcal{C}(\mathcal{V}_h, \epsilon)$ with size $|\mathcal{C}(\mathcal{V}_h, \epsilon)| \leq N_{\mathcal{V}_h}(\epsilon)$. For any $\epsilon > 0$, we define the uniform covering number of $\mathcal{V}$ with respect to ϵ as $N_{\mathcal{V}}(\epsilon) := \max_{h \in [H]} N_{\mathcal{V}_h}(\epsilon)$.

4 Algorithm

In this section, we introduce our algorithm `GFA-RFE` as presented in Algorithm 1. `GFA-RFE` consists of two phases, where in the first exploration phase, `GFA-RFE` collects K episodes without reward signal. Then in the second planning phase, `GFA-RFE` leverages the collected K episodes to learn a policy trying to maximize the cumulative reward given a specific reward function r. The details of these two phases are presented in the following subsections.

4.1 Exploration Phase: Efficient Exploration via Uncertainty-aware Intrinsic Reward

The ultimate goals of the exploration phase are exploring environments and collecting data in the absence of reward to facilitate finding the near-optimal policy in the next phase. At a high level, GFA−RFE achieves these goals by encouraging the agent to explore regions containing higher uncertainty, which intuitively guarantees the maximal information gained in each episode.

Intrinsic reward. GFA−RFE evaluate the uncertainty by $D_{\mathcal{F}_h}$ in Definition 3.4, and uses its oracle $\overline{D}_{\mathcal{F}_h}$ as the intrinsic reward $r_{k,h}$ in Line 1 to generate an uncertainty-target policy in Line 1. Recall that $D^2_{\mathcal{F}_h}(z; z_{[k-1],h}, \sigma_{[k-1],h})$ is defined as

$$\sup_{f_1,f_2\in\mathcal{F}_h} \frac{(f_1(z)-f_2(z))^2}{\sum_{s\in[i-1]} \frac{1}{\sigma^2_{s,h}}(f_1(z_{s,h})-f_2(z_{s,h}))^2+\lambda}.$$

In particular, a high reward signal means that there exist functions in $\mathcal{F}_h$ close to each other on all historical observations but divergent for the current state and action pair. This further suggests that the past observations are not enough for the agent to make a precise value estimation for the current state-action pair.

Weighted regression. The usage of the intrinsic reward $r_{k,h}$ induces an intrinsic action-value function $Q^*_{k,h}(\cdot,\cdot;r_k)$, which serves as a metric for cumulative uncertainty of remaining stages. As in model-free approaches, GFA−RFE aims to estimate $Q^*_{k,h}(\cdot,\cdot;r_k)$ and further finds a policy π^k_h that would maximize the cumulative uncertainty over H stages. This part is presented in Algorithm 1 through Line 1 to Line 1.

To reduce the estimation error, GFA−RFE incorporates the weighted regression proposed in Zhao et al. (2023) into estimating $Q^*_{k,h}(s,a;r_k)$. The algorithm starts at final stage $h=H$ and estimating the $Q^*_{k,h}(s,a;r_k)$ approximated by function $\widehat{f}_{k,h}$ using Bellman equation:

$$\begin{aligned}\widehat{f}_{k,h}(s_h,a_h) &= r_{k,h}(s_h,a_h)+[\mathbb{P}_hV_{k,h+1}](s_h,a_h)\\ &\approx r_{k,h}(s_h,a_h)+V_{k,h+1}(s_{h+1}).\end{aligned}$$

However, estimating $[\mathbb{P}_hV_{k,h+1}](s_h,a_h)$ using $V_{k,h+1}(s_{h+1})$ may also introduce error since the variance of distribution $\mathbb{P}_h(\cdot|s,a)$ varies among different state-action pair. Therefore, we tackle this heterogeneous variance issue by minimizing the Bellman residual loss weighted by using the estimated variance $\bar{\sigma}_{k,h}$ of observed state-action pairs s^i_h, a^i_h:

$$\sum_{i\in[k-1]} \frac{(f_{i,h}(s^i_h,a^i_h)-r_{i,h}(s^i_h,a^i_h)-V_{i,h+1}(s^i_{h+1}))^2}{\bar{\sigma}^2_{i,h}}.$$

Obviously, a lower variance $\bar{\sigma}_{i,h}$ yields a larger weight during the regression. The calculation of variances $\bar{\sigma}_{i,h}$ involves both *aleatoric uncertainty* and *epistemic uncertainty* (Kendall & Gal, 2017; Mai et al., 2022), where the *aleatoric uncertainty* is $\sigma_{k,h}$ calculated in Line 1 caused by indeterminism of the transition and *epistemic uncertainty* is $\overline{D}^{1/2}_{\mathcal{F}_h}$ caused by limited data. Such an approach can be proved to improve the sample efficiency of our algorithm GFA−RFE (see Theorem 5.1 and its discussion). Similar approaches have been used in Zhou et al. (2021); Ye et al. (2023) to provide more robust and efficient estimation.

After obtaining the $\widehat{f}_{k,h}$ function through weighted regression, GFA−RFE follows the standard optimism design in online exploration methods to add the bonus term $b_{k,h}$ for overestimating the $Q^*_{k,h}(s,a;r)$ function in Line 1. Using this optimistic estimation, GFA−RFE thus takes the greedy policy and estimates the value function $V_{k,h}$ in Line 1 before proceeding to the previous stage $h-1$.

4.2 Planning Phase: Effective Planning Using Weighted Regression

After exploring environments and collecting data in the exploration phase, the agent is now given the reward for a specific task, but no longer interacts with the environment. GFA−RFE enters its planning phase and ensures a policy to maximize the cumulative reward of r_h across all H stages. GFA−RFE estimates $Q^*_h(s,a;r)$ by weighted regression and further finds the optimal policy π_h, which is the same process as in the exploration phase. This part is presented in Algorithm1 through Line 1 to Line 1.

Remark 4.1. Compared with Kong et al. (2021), our algorithm leverages the advantage of generalized elude dimension and incorporates the estimated variance σ into 1) weighted regression in Line 1 in the planning phase and Line 1 in exploration phase; 2) intrinsic reward design in Line 1. Also, our algorithm does not set the reward $r_{k,h}=b_{k,h}/H$ as of Kong et al. (2021); Wang et al. (2020), thus the agent can explore more aggressively and more efficiently using the knowledge of variance of the observation. Therefore, GFA−RFE is more sample efficient compared with Kong et al. (2021), which is discussed in detail in Remark 5.7.

5 Theoretical Results

We analyze GFA−RFE theoretically in this section. The uncertainty-aware reward-free exploration mechanism leads to efficient learning with provable sample complexity guarantees. The first theorem characterizes how the suboptimality decays as exploration time grows.

Theorem 5.1. *For GFA−RFE, set confidence radius $\beta^E = \widetilde{O}\big(\sqrt{H\log N_{\mathcal{V}}(\epsilon)}\big)$ and $\beta^P = \widetilde{O}\big(\sqrt{H\log N_{\mathcal{F}}(\epsilon)}\big)$, and take $\alpha = 1/\sqrt{H}$ and $\gamma = \sqrt{\log N_{\mathcal{V}}(\epsilon)}$. Then, for any $\delta\in(0,1)$, with probability at least $1-\delta$, after collecting K episodes of samples, for any reward function $r=\{r_h\}_{h=1}^H$ such that $\sum_{h=1}^H r_h(s_h,a_h)\le 1$, GFA−RFE outputs a policy*

Algorithm 1 GFA-RFE

Require: Confidence radius β^E
Require: Regularization parameter λ
Phase I: Exploration Phase
for $k = 1, 2, \cdots, K$ **do**
  for $h = H, H-1, \cdots, 1$ **do**
    $b_{k,h}(\cdot,\cdot) \leftarrow 2\beta^E \cdot \overline{\mathcal{D}}_{\mathcal{F}_h}(\cdot,\cdot; z_{[k-1],h}, \bar{\sigma}_{[k-1],h})$.
    $r_{k,h}(\cdot,\cdot) \leftarrow b_{k,h}(\cdot.\cdot)/2$.
    $\widehat{f}_{k,h} \leftarrow \operatorname{argmin}_{f_h \in \mathcal{F}_h} \sum_{i\in[k-1]} \frac{1}{\bar{\sigma}^2_{i,h}} (f_h(s^i_h, a^i_h) - r_{k,h}(s^i_h, a^i_h) - V_{k,h+1}(s^i_{h+1}))^2$.
    $Q_{k,h}(s,a) \leftarrow \min\Big\{\widehat{f}_{k,h}(s,a) + b_{k,h}(s,a), 1\Big\}$.
    $V_{k,h}(s) \leftarrow \max_a Q_{k,h}(s,a)$.
    Set the policy $\pi^k_h(\cdot) \leftarrow \operatorname{argmax}_{a\in\mathcal{A}} Q_{k,h}(\cdot, a)$.
  end for
  Receive the initial state s^k_1.
  for stage $h = 1, \ldots, H$ **do**
    Take action $a^k_h \leftarrow \pi^k_h(s^k_h)$, receive next state s^k_{h+1}.
    $\sigma_{k,h} \leftarrow 2\sqrt{\log N_{\mathcal{V}}(\epsilon) \cdot \min\{\widehat{f}_{k,h}(s^k_h, a^k_h), 1\}}$.
    $\bar{\sigma}_{k,h} \leftarrow \max\Big\{\gamma \cdot \overline{D}^{1/2}_{\mathcal{F}_h}(z_{k,h}; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}), \sigma_{k,h}, \alpha\Big\}$.
  end for
end for
Phase II: Planning Phase
Require: Dataset $\{(s^k_h, a^k_h, \bar{\sigma}^2_{k,h})\}_{(k,h)\in[K]\times[H]}$
Require: Confidence radius β^P
Require: Reward function $r = \{r_h\}_{h\in[H]}$
Initiate $\widehat{V}_{H+1}(\cdot) \leftarrow 0$, $\widehat{Q}_{H+1}(\cdot,\cdot) \leftarrow 0$
for step $h = H, \cdots, 1$ **do**
  $b_h(\cdot,\cdot) \leftarrow \min\{\beta^P \overline{\mathcal{D}}_{\mathcal{F}_h}(z; z_{[K],h}, \bar{\sigma}_{[K],h}), 1\}$.
  $\widehat{f}_h \leftarrow \operatorname{argmin}_{f_h\in\mathcal{F}_h} \sum_{i\in[K]} \frac{1}{\bar{\sigma}^2_{i,h}} (f_h(s^i_h, a^i_h) - r_h(s^i_h, a^i_h) - \widehat{V}_{h+1}(s^i_{h+1}))^2$.
  $\widehat{Q}_h(s,a) \leftarrow \min\Big\{\widehat{f}_h(s,a) + b_h(s,a), 1\Big\}$.
  $\widehat{V}_h(\cdot) \leftarrow \max_{a\in\mathcal{A}} \widehat{Q}_h(\cdot, a)$.
  $\pi_h(\cdot) \leftarrow \operatorname{argmax}_{a\in\mathcal{A}} \widehat{Q}_h(\cdot, a)$.
end for
Ensure: Policy π

π satisfying the following sub-optimality bound,

$$\mathbb{E}_{s_1\sim\mu}[V^*_1(s_1; r) - V^\pi_1(s_1; r)] = \widetilde{O}\Big(H\sqrt{\log N_{\mathcal{F}}(\epsilon)}\sqrt{\dim_{\alpha,K}(\mathcal{F})/K}\Big).$$

We are now ready to present the sample complexity of GFA-RFE for the reward-free exploration.

Corollary 5.2. *Under the same conditions in Theorem 5.1, with probability at least $1-\delta$, for any reward function $r = \{r_h\}_{h=1}^H$ such that $\sum_{h=1}^H r_h(s_h, a_h) \le 1$, GFA-RFE returns an ϵ-optimal policy after collecting $K \le \widetilde{\mathcal{O}}(H^2 \log N_{\mathcal{F}}(\epsilon) \dim_{\alpha,K}(\mathcal{F})\epsilon^{-2})$ episodes during the exploration phase.*

Remark 5.3. Let $d_{K,\delta}$ be $\max\{\log N_{\mathcal{F}}(\epsilon), \dim_{\alpha,K}(\mathcal{F})\}$, GFA-RFE yields an $\widetilde{\mathcal{O}}(H^2 d^2_{K,\delta}\epsilon^{-2})$ sample complexity for reward-free exploration with high probability. In tabular setting, $d_{K,\delta} = \widetilde{\mathcal{O}}(SA)$, thus yields an $\widetilde{\mathcal{O}}(H^2S^2A^2\epsilon^{-2})$ sample complexity. In linear MDPs and generalized linear MDPs with dimension d, $d_{K,\delta} = \widetilde{\mathcal{O}}(d)$, thus yields an $\widetilde{\mathcal{O}}(H^2d^2\epsilon^{-2})$ sample complexity which matches the result from Hu et al. (2022). For a more general setting where the function class with eluder dimension d, $d_{K,\delta} = \widetilde{\mathcal{O}}(d)$, which yields a $\widetilde{\mathcal{O}}(H^2d^2\epsilon^{-2})$ sample complexity.

For a fair comparison with some existing works, we translate our sample complexity result to the case where the reward scale is $r_h \in [0,1]$, $\forall h \in [H]$. The result can be trivially obtained by replacing $r \to r/H$ in GFA-RFE.

Corollary 5.4. *With probability at least $1-\delta$, for any reward function such that $r_h(s,a) \in [0,1]$ or the total reward is bounded by $\sum_{h=1}^H r_h(s_h, a_h) \le H$, GFA-RFE returns an ϵ-optimal policy after collecting $K \le \widetilde{\mathcal{O}}(H^4 d^2_{K,\delta}\epsilon^{-2})$ episodes in the exploration phase.*

Remark 5.5. Compared with Chen et al. (2022a) which provides a $\widetilde{\mathcal{O}}(d \log|\mathcal{P}|\epsilon^{-2})$ sample complexity for model-based RL, GFA-RFE is a *model-free* algorithm which does not need to directly sample transition kernel $\mathbb{P}_h(\cdot|\cdot,\cdot)$ from all possible transitions $\widetilde{\Delta}(\Pi)$, therefore, GFA-RFE is computationally efficient and can be easily implemented based on the current empirical DRL algorithms.

Remark 5.6. Compared with Chen et al. (2022b) which achieves a $\widetilde{\mathcal{O}}(H^7d^3\epsilon^{-2})$ sample complexity, one can find our result significantly improves the dependency on H, d. Chen et al. (2022b) didn't optimize the exploration policy by constructing intrinsic rewards but by updating Bellman error constraints on the value function class. It sacrificed the sample complexity to adapt the general function approximation settings. In addition, this approach is generally computationally intractable as it explicitly maintains feasible function classes. For its V-type variant, it even maintains a finite cover of the function class, which can be exponentially large.

Remark 5.7. Kong et al. (2021) leveraged the "sensitivity" as the intrinsic reward during the exploration and achieved a $\widetilde{\mathcal{O}}(H^6d^4\epsilon^{-2})$ reward-free sample complexity. Compare their algorithm and ours, ours improves a H^2d^2 factor from 1) using weighted regression to handle heterogeneous observations 2) using a "truncated Bellman equation" (Chen et al., 2021) in our analysis, and 3) a properly improved uncertainty metric $\overline{D}^2_{\mathcal{F}_h}$ instead of the sensitivity.

6 Experiments

6.1 Experiment Setup

Based on our theoretical perceptive, we integrate our algorithm in the unsupervised reinforcement learning (URL) framework and evaluate the performance of the proposed

Table 2. Cumulative reward for various exploration algorithms across different environments and tasks. The cumulative reward is averaged over 8 individual runs for both online exploration and offline planning. The result for each individual run is obtained by evaluating the policy network using the last-iteration parameter. Standard deviation is calculated across these runs. Results presented in **boldface** denote the best performance for each task, and those underlined represent the second-best outcomes. The cyan background highlights results of our algorithms.

Environment	Task	Baselines							Ours
		ICM	APT	DIAYN	APS	Dis.	SMM	RND	GFA-RFE
Walker	Flip	177 ± 80	523 ± 57	207 ± 119	246 ± 103	$\mathbf{570 \pm 32}$	242 ± 71	507 ± 48	554 ± 64
	Run	108 ± 41	304 ± 38	113 ± 38	132 ± 39	$\mathbf{340 \pm 37}$	116 ± 21	306 ± 34	339 ± 34
	Stand	466 ± 17	891 ± 62	587 ± 169	573 ± 177	726 ± 79	443 ± 104	750 ± 62	$\mathbf{925 \pm 50}$
	Walk	411 ± 237	772 ± 60	432 ± 222	645 ± 156	$\mathbf{851 \pm 63}$	273 ± 162	709 ± 115	826 ± 89
Quadruped	Run	93 ± 68	452 ± 49	158 ± 64	159 ± 82	$\mathbf{524 \pm 24}$	162 ± 140	522 ± 30	460 ± 36
	Jump	89 ± 47	740 ± 91	218 ± 114	123 ± 67	$\mathbf{829 \pm 22}$	211 ± 127	790 ± 38	719 ± 68
	Stand	207 ± 134	910 ± 45	331 ± 81	308 ± 147	$\mathbf{953 \pm 16}$	239 ± 104	940 ± 27	867 ± 61
	Walk	94 ± 60	680 ± 117	171 ± 72	141 ± 80	720 ± 175	125 ± 36	$\mathbf{820 \pm 94}$	726 ± 146

algorithm in URL benchmark (Laskin et al., 2021).[2] As suggested by Ye et al. (2023), we use the variance of n-ensembled Q functions as the estimation of the bonus oracle $\overline{D}^2_{\mathcal{F}}$ which will be used in (1) intrinsic reward $r_{k,h}$; (2) exploration bonus $b_{k,h}$; and (3) weights $\sigma^2_{k,h}$ for the value target regression. All these Q networks are trained by Q-learning with different mini-batches in the replay buffer. Obviously, the variance of these Q networks comes from the randomness of initialization and the randomness of different mini-batches used in training. The pseudo code for the practical algorithm is deferred to Appendix F.

The original implementation of Laskin et al. (2021) involves two phases where the neural network is first *pretrained* by interacting with the environment without receiving reward signals and then *finetuned* by interacting with the environment again with reward signal. However, in our experiments, we strictly follow the design of reward-free exploration by first exploring the environment without the reward. The explored trajectories are collected into a dataset $\mathcal{D} = \{(s, a, s')\}$. Then we call a reward oracle r to assign rewards to this dataset $\mathcal{D}$ and learn the optimal policy using the offline dataset $\mathcal{D}_r = \{(s, a, s', r(s, a, s'))\}$ without interacting the dataset anymore. Intuitively speaking, this *online exploration* + *offline planning* paradigm is more challenging than the *online pretraining* + *online finetuning* and would be more practical, especially with different reward signals.

Unsupervised reinforcement learning benchmarks. We conduct our experiments on *Unsupervised Reinforcement Learning Benchmarks* (Laskin et al., 2021), which consists of two multi-tasks environments (*Walker*, *Quadruped*) from DeepMind Control Suite (Tunyasuvunakool et al., 2020). Each environment is equipped with several reward functions and goals. For example, *Walker-run* consists of rewards encouraging the walker to run at speed and *Walker-stand* consists of rewards indicating the walker should stead steadily. We consider the state-based input in our experiments where the agent can directly observe the current state instead of image inputs (a.k.a. pixel-based).

Baseline algorithms. We inherit the baseline algorithms ICM (Pathak et al., 2017), Disagreement (Pathak et al., 2019), RND (Burda et al., 2018b), APT (Liu & Abbeel, 2021b), DIAYN (Eysenbach et al., 2018), APS (Liu & Abbeel, 2021a), SMM (Lee et al., 2019). All these algorithms provide different 'intrinsic rewards' in place of ours during exploration. We make all these baseline algorithms align with our settings which first collect an exploration dataset and then do offline training on the collected dataset with rewards.

6.2 Experiment Results

Experimental results are presented in Table 2. It's obvious that GFA-RFE can efficiently explore the environment without the reward function and then output a near-optimal policy given various reward functions. For the baseline algorithms, APT, Disagreement, and RND perform consistently better than the rest of the 4 algorithms on all 2 environments and 8 tasks. The performance of GFA-RFE enjoys compatible or superior performance compared with these top-level methods (APT, Disagreement, and RND), on these tasks. These promising numerical results justify our theoretical results and show that GFA-RFE can indeed efficiently learn the environment in a practical setting.

Ablation study. To verify the performance of our algorithm, we also did ablation studies on 1) the relationship between offline training processes and episodic reward 2) the relationship between the quantity of online exploration data used in offline training and the achieve episodic reward. The details of the ablation study are deferred to Appendix F.

[2] Our implementation can be accessed at GitHub via https://github.com/uclaml/GFA-RFE.

7 Conclusion

We study reward-free exploration under general function approximation in this paper. We show that, with an uncertainty-aware intrinsic reward and variance-weighted regression on learning the environment, `GFA-RFE` can be theoretically proved to explore the environment efficiently without the existence of reward signals. Experiments show that our design of intrinsic reward can be efficiently implemented and effectively used in an unsupervised reinforcement learning paradigm. In addition, experiment results verify that adding uncertainty estimation to the learning processes can improve the sample efficiency of the algorithm, which is aligned with our theoretical results of weighted regression.

Acknowledgements

We thank the anonymous reviewers for their helpful comments. WZ is supported by the UCLA Dissertation Year Fellowship. QG is supported in part by the National Science Foundation CAREER Award 1906169 and research fund from UCLA-Amazon Science Hub. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

Impact Statement

This paper discusses efforts aimed at enhancing the domain of reinforcement learning, encompassing both theoretical insights and empirical findings. Our approaches are poised to offer beneficial societal effects as they circumvent the reliance on pre-defined reward functions, which could embed biases and prejudicial assessments. Furthermore, our robust theoretical foundation enhances the transparency of exploration processes within reinforcement learning frameworks. While our algorithm potentially improve the accountability of exploration of RL, it does not eliminate the risks of unsafe behavior or offending to the constrain of exploration, which might be addressed in the future works. Other than that we do not see any potential negative impacts of our work that need to be specifically pointed out here, to the best of our knowledge.

References

Agarwal, A., Jin, Y., and Zhang, T. Voql: Towards optimal regret in model-free rl with nonlinear function approximation. *arXiv preprint arXiv:2212.06069*, 2022.

Burda, Y., Edwards, H., Pathak, D., Storkey, A., Darrell, T., and Efros, A. A. Large-scale study of curiosity-driven learning. *arXiv preprint arXiv:1808.04355*, 2018a.

Burda, Y., Edwards, H., Storkey, A., and Klimov, O. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018b.

Chen, F., Mei, S., and Bai, Y. Unified algorithms for rl with decision-estimation coefficients: No-regret, pac, and reward-free learning. *arXiv preprint arXiv:2209.11745*, 2022a.

Chen, J., Modi, A., Krishnamurthy, A., Jiang, N., and Agarwal, A. On the statistical efficiency of reward-free exploration in non-linear rl. *Advances in Neural Information Processing Systems*, 35:20960–20973, 2022b.

Chen, X., Hu, J., Yang, L., and Wang, L. Near-optimal reward-free exploration for linear mixture mdps with plug-in solver. In *International Conference on Learning Representations*, 2021.

Chen, Z., Li, C. J., Yuan, A., Gu, Q., and Jordan, M. I. A general framework for sample-efficient function approximation in reinforcement learning. *arXiv preprint arXiv:2209.15634*, 2022c.

Eysenbach, B., Gupta, A., Ibarz, J., and Levine, S. Diversity is all you need: Learning skills without a reward function. *arXiv preprint arXiv:1802.06070*, 2018.

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

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

Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., and Davidson, J. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019b.

He, K., Zhang, X., Ren, S., and Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

Hu, P., Chen, Y., and Huang, L. Towards minimax optimal reward-free reinforcement learning in linear mdps. In *The Eleventh International Conference on Learning Representations*, 2022.

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

Jin, C., Krishnamurthy, A., Simchowitz, M., and Yu, T. Reward-free exploration for reinforcement learning. In *International Conference on Machine Learning*, pp. 4870–4879. PMLR, 2020a.

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

Jin, C., Liu, Q., and Miryoosefi, S. Bellman eluder dimension: New rich classes of rl problems, and sample-efficient algorithms. *Advances in neural information processing systems*, 34:13406–13418, 2021.

Kaufmann, E., Ménard, P., Domingues, O. D., Jonsson, A., Leurent, E., and Valko, M. Adaptive reward-free exploration. In *Algorithmic Learning Theory*, pp. 865–891. PMLR, 2021.

Kendall, A. and Gal, Y. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017.

Kong, D., Salakhutdinov, R., Wang, R., and Yang, L. F. Online sub-sampling for reinforcement learning with general function approximation. *arXiv preprint arXiv:2106.07203*, 2021.

Laskin, M., Srinivas, A., and Abbeel, P. Curl: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning*, pp. 5639–5650. PMLR, 2020.

Laskin, M., Yarats, D., Liu, H., Lee, K., Zhan, A., Lu, K., Cang, C., Pinto, L., and Abbeel, P. Urlb: Unsupervised reinforcement learning benchmark. *arXiv preprint arXiv:2110.15191*, 2021.

Lee, L., Eysenbach, B., Parisotto, E., Xing, E., Levine, S., and Salakhutdinov, R. Efficient exploration via state marginal matching. *arXiv preprint arXiv:1906.05274*, 2019.

Levine, S., Finn, C., Darrell, T., and Abbeel, P. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

Liu, H. and Abbeel, P. Aps: Active pretraining with successor features. In *International Conference on Machine Learning*, pp. 6736–6747. PMLR, 2021a.

Liu, H. and Abbeel, P. Behavior from the void: Unsupervised active pre-training. *Advances in Neural Information Processing Systems*, 34:18459–18473, 2021b.

Mai, V., Mani, K., and Paull, L. Sample efficient deep reinforcement learning via uncertainty estimation. *arXiv preprint arXiv:2201.01666*, 2022.

Ménard, P., Domingues, O. D., Jonsson, A., Kaufmann, E., Leurent, E., and Valko, M. Fast active learning for pure exploration in reinforcement learning. In *International Conference on Machine Learning*, pp. 7599–7608. PMLR, 2021.

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

Papini, M., Tirinzoni, A., Pacchiano, A., Restelli, M., Lazaric, A., and Pirotta, M. Reinforcement learning in linear mdps: Constant regret and representation selection. *Advances in Neural Information Processing Systems*, 34: 16371–16383, 2021.

Pathak, D., Agrawal, P., Efros, A. A., and Darrell, T. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pp. 2778–2787. PMLR, 2017.

Pathak, D., Gandhi, D., and Gupta, A. Self-supervised exploration via disagreement. In *International conference on machine learning*, pp. 5062–5071. PMLR, 2019.

Qiu, S., Wang, L., Bai, C., Yang, Z., and Wang, Z. Contrastive ucb: Provably efficient contrastive self-supervised learning in online reinforcement learning. In *International Conference on Machine Learning*, pp. 18168–18210. PMLR, 2022.

Russo, D. and Van Roy, B. Eluder dimension and the sample complexity of optimistic exploration. In *NIPS*, pp. 2256–2264. Citeseer, 2013.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

Stooke, A., Lee, K., Abbeel, P., and Laskin, M. Decoupling representation learning from reinforcement learning. In *International Conference on Machine Learning*, pp. 9870–9879. PMLR, 2021.

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

Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., Casas, D. d. L., Budden, D., Abdolmaleki, A., Merel, J., Lefrancq, A., et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

Tunyasuvunakool, S., Muldal, A., Doron, Y., Liu, S., Bohez, S., Merel, J., Erez, T., Lillicrap, T., Heess, N., and Tassa, Y. dm_control: Software and tasks for continuous control. *Software Impacts*, 6:100022, 2020. ISSN 2665-9638. doi: https://doi.org/10.1016/j.simpa.2020.100022.

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

Wagenmaker, A. J., Chen, Y., Simchowitz, M., Du, S., and Jamieson, K. Reward-free rl is no harder than reward-aware rl in linear markov decision processes. In *International Conference on Machine Learning*, pp. 22430–22456. PMLR, 2022.

Wang, R., Du, S. S., Yang, L. F., and Salakhutdinov, R. On reward-free reinforcement learning with linear function approximation. *Advances in neural information processing systems*, 2020.

Yarats, D., Fergus, R., Lazaric, A., and Pinto, L. Reinforcement learning with prototypical representations. In *International Conference on Machine Learning*, pp. 11920–11931. PMLR, 2021a.

Yarats, D., Zhang, A., Kostrikov, I., Amos, B., Pineau, J., and Fergus, R. Improving sample efficiency in model-free reinforcement learning from images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 10674–10681, 2021b.

Ye, C., Yang, R., Gu, Q., and Zhang, T. Corruption-robust offline reinforcement learning with general function approximation. *arXiv preprint arXiv:2310.14550*, 2023.

Zanette, A., Lazaric, A., Kochenderfer, M. J., and Brunskill, E. Provably efficient reward-agnostic navigation with linear value iteration. *Advances in Neural Information Processing Systems*, 2020.

Zhang, J., Zhang, W., and Gu, Q. Optimal horizon-free reward-free exploration for linear mixture mdps. *arXiv preprint arXiv:2303.10165*, 2023.

Zhang, W., He, J., Zhou, D., Zhang, A., and Gu, Q. Provably efficient representation learning in low-rank markov decision processes. *arXiv preprint arXiv:2106.11935*, 2021a.

Zhang, W., Zhou, D., and Gu, Q. Reward-free model-based reinforcement learning with linear function approximation. *Advances in Neural Information Processing Systems*, 34, 2021b.

Zhang, Z., Du, S. S., and Ji, X. Nearly minimax optimal reward-free reinforcement learning. *arXiv preprint arXiv:2010.05901*, 2020.

Zhang, Z., Zhou, Y., and Ji, X. Model-free reinforcement learning: from clipped pseudo-regret to sample complexity. In *International Conference on Machine Learning*, pp. 12653–12662. PMLR, 2021c.

Zhao, H., He, J., and Gu, Q. A nearly optimal and low-switching algorithm for reinforcement learning with general function approximation. *arXiv preprint arXiv:2311.15238*, 2023.

Zhou, D., Gu, Q., and Szepesvari, C. Nearly minimax optimal reinforcement learning for linear mixture markov decision processes. In *Conference on Learning Theory*, pp. 4532–4576. PMLR, 2021.

A Proof of Theorems in Section 5

A.1 Additional Definitions and High Probability Events

In this section, we introduce additional definitions that will be used in the proofs. Also, we define the good events that GFA−RFE is guaranteed to have near-optimal sample complexity.

Definition A.1 (Truncated Optimal Value Function)**.** We define the following truncated value functions for any reward r:

$$\begin{aligned}\widetilde{V}^*_{H+1}(s;r) &= 0, \quad \forall s \in \mathcal{S}\\ \widetilde{Q}^*_h(s,a;r) &= \min\{r_h(s,a) + \mathbb{P}_h \widetilde{V}^*_{h+1}(s,a;r), 1\}, \quad \forall (s,a) \in \mathcal{S}\times\mathcal{A}\\ \widetilde{V}^*_h(s;r) &= \max_{a\in\mathcal{A}} \widetilde{Q}^*_h(s,a;r). \quad \forall s \in \mathcal{S}, h \in [H].\end{aligned}$$

The good event $\mathcal{E}^E_{k,h}$ at stage h of episode k in exploration phase is defined to be:

$$\mathcal{E}^E_{k,h} = \Big\{\lambda + \sum_{i\in[k-1]} \frac{1}{\bar{\sigma}^2_{i,h}} \Big(\widehat{f}_{k,h}(s^i_h, a^i_h) - \mathcal{T}_h V_{k,h+1}(s^i_h, a^i_h)\Big)^2 \le (\beta^E)^2\Big\}.$$

The intersection of all good events in exploration phase is:

$$\mathcal{E}^E := \bigcap_{k\ge 1, h\in[H]} \mathcal{E}^E_{k,h}.$$

The following lemma indicates that $\mathcal{E}$ holds with high probability for GFA−RFE.

Lemma A.2. *In Algorithm 1, for any $\delta \in (0,1)$ and fixed $h \in [H]$, with probability at least $1-\delta$, $\mathcal{E}^E$ holds.*

In the planning phase, we define the good events for exploration phase with indicator functions as

$$\begin{aligned}\overline{\mathcal{E}}^P_h &= \Big\{\lambda + \sum_{i\in[K]} \frac{\mathbb{1}_h}{\bar{\sigma}^2_{i,h}} \Big(\widehat{f}_h(s^i_h, a^i_h) - \mathcal{T}_h \widehat{V}_{h+1}(s^i_h, a^i_h)\Big)^2 \le (\widehat{\beta}^P)^2\Big\},\\ \overline{\mathcal{E}}^P &= \bigcap_{h\in[H]} \overline{\mathcal{E}}^P_h,\end{aligned}$$

where $\widehat{\mathbb{1}}_h = \mathbb{1}(V^*_{h+1}(s) \le \widehat{V}_{h+1}(s),\ \forall s \in \mathcal{S}) \cdot \mathbb{1}(\widehat{V}_{h+1}(s) \le V_{k,h+1}(s) + V^*(s;r),\ \forall s \in \mathcal{S}) \cdot \mathbb{1}([\mathbb{V}_h(\widehat{V}_{h+1} - V^*_{h+1})](s^k_h, a^k_h) \le \eta^{-1}\bar{\sigma}^2_{k,h},\ \forall k \in [K])$ and $\eta = \log N_{\mathcal{V}}(\epsilon)$. Like in the exploration phase, we also have that $\overline{\mathcal{E}}^P$ holds with high probability for GFA−RFE.

Lemma A.3. *In Algorithm 1, for any $\delta \in (0,1)$ and fixed $h \in [H]$, with probability at least $1-\delta$, $\overline{\mathcal{E}}^P$ holds.*

Furthermore, we have the following good events in the planning phase without indicator function:

$$\mathcal{E}^P_h = \Big\{\lambda + \sum_{i=1}^{k-1} \frac{1}{(\bar{\sigma}_{i,h'})^2} \Big(\widehat{f}_{h'}(s^i_{h'}, a^i_{h'}) - \mathcal{T}_{h'} V_{h'+1}(s^i_{h'}, a^i_{h'})\Big)^2 \le (\beta^P)^2,\ \forall h \le h' \le H, k \in [K]\Big\}.$$

And we define $\mathcal{E}^P := \mathcal{E}^P_1$. We shows that $\mathcal{E}^P$ holds if both $\mathcal{E}^E, \overline{\mathcal{E}}^P$ hold with the help of the following lemma:

Lemma A.4. *If the event $\mathcal{E}^E, \overline{\mathcal{E}}^P, \mathcal{E}^P_{h+1}$ all hold, then event $\mathcal{E}^P_h$ holds.*

Since $\mathcal{E}^P_H$ holds trivially, Lemma A.4 indicates that $\mathcal{E}^P$ holds.

A.2 Covering Number

The optimistic value functions at stage $h \in [H]$ in our construction belong to the following function class:

$$\mathcal{V}_h = \left\{V(\cdot) = \max_{a\in\mathcal{A}} \min\left(1, f(\cdot,a) + \beta \cdot b(\cdot,a)\right) \middle| f \in \mathcal{F}_h, b \in \mathcal{B}\right\}. \tag{A.1}$$

Lemma A.5 (ϵ-covering number of optimistic value function classes)**.** *For optimistic value function class $\mathcal{V}_{k,h}$ defined in (A.1), we define the distance between two value functions V_1 and V_2 as $\|V_1 - V_2\|_\infty := \max_{s\in\mathcal{S}} |V_1(s) - V_2(s)|$. Then the ϵ-covering number with respect to the distance function can be upper bounded by*

$$N_{\mathcal{V}_h}(\epsilon) := N_{\mathcal{F}_h}(\epsilon/2) \cdot N_{\mathcal{B}}(\epsilon/2\beta). \tag{A.2}$$

Lemma A.5 further indicates that

$$N_{\mathcal{V}}(\epsilon) = \max_{h\in[H]} N_{\mathcal{V}_h}(\epsilon) = \max_{h\in[H]} N_{\mathcal{F}_h}(\epsilon/2)\cdot N_{\mathcal{B}}(\epsilon/2\beta) = N_{\mathcal{F}}(\epsilon/2)\cdot N_{\mathcal{B}}(\epsilon/2\beta).$$

A.3 Proof of Theorems

We first introduce the following lemmas to build the path to Theorem 5.1.

Lemma A.6. *On the event $\mathcal{E}^P$, we have*

$$|\widehat{f}_h(s,a) - \mathcal{T}_h\widehat{V}_{h+1}| \le \beta^P D_{\mathcal{F}_h}(z; z_{[K],h}, \bar{\sigma}_{[K],h}).$$

Lemma A.7 (Optimism in the planning phase)**.** *On the event $\mathcal{E}^P$, for any $h\in[H]$, we have*

$$V_h^*(s;r) \le \widehat{V}_h(s), \quad \forall s\in\mathcal{S}.$$

Lemma A.8. *On the event $\underline{\mathcal{E}}^E$, with probability at least $1-3\delta$, we have*

$$\sum_{k=1}^K V_{k,1}(s_1^k) = O(\beta^E\sqrt{\dim_{\alpha,K}(\mathcal{F})}H\sqrt{K}).$$

Lemma A.9. *With probability $1-\delta$, we have*

$$\Big|\sum_{k=1}^K \big(\mathbb{E}_{s\sim\mu}\big[\widetilde{V}_1^*(s;r_k)\big] - \widetilde{V}_1^*(s;r_k)\big)\Big| \le \sqrt{2K\log(1/\delta)}.$$

We denote the event that Lemma A.8 holds as Φ, and the event that Lemma A.9 holds as Ψ.

Lemma A.10. *Under event $\mathcal{E}^E\cap\Phi\cap\Psi$, we have*

$$\mathbb{E}_{s\sim\mu}\Big[\widetilde{V}_1^*(s;b)\Big] = O\Big(\beta^E\sqrt{H\dim_{\alpha,K}(\mathcal{F})/K}\sqrt{\log N_{\mathcal{F}}(\epsilon)/\log N_{\mathcal{V}}(\epsilon)}\Big),$$

where $b=\{b_h\}_{h=1}^H$ is the UCB bonus in planning phase.

With these lemmas, we are ready to prove Theorem 5.1.

Proof of Theorem 5.1. By Lemma A.7, we can upper bound the suboptimality as

$$\mathbb{E}_{s_1\sim\mu}[V_1^*(s_1;r) - V_1^\pi(s_1;r)] \le \mathbb{E}_{s_1\sim\mu}[\widehat{V}_1(s_1) - V_1^\pi(s_1;r)].$$

Then, we can decompose the difference between optimistic estimate of value function and the true value function in the following:

$$\begin{aligned}
&\mathbb{E}_{s_1\sim\mu}[\widehat{V}_1(s_1) - V_1^\pi(s_1;r)]\\
&= \mathbb{E}_{s_1\sim\mu}\Big[\min\{\widehat{f}_1(s_1,\pi(s_1)) + b_1(s_1,\pi(s_1)),1\} - r_1(s_1,\pi(s_1)) - \mathbb{P}_1V_2^\pi(s_1,\pi(s_1);r)\Big]\\
&\le \mathbb{E}_{s_1\sim\mu}\Big[\min\{\widehat{f}_1(s_1,\pi(s_1)) + b_1(s_1,\pi(s_1)) - r_1(s_1,\pi(s_1)) - \mathbb{P}_1V_2^\pi(s_1,\pi(s_1);r),1\}\Big]\\
&= \mathbb{E}_{s_1\sim\mu}\Big[\min\Big\{\widehat{f}_1(s_1,\pi(s_1)) - r_1(s_1,\pi(s_1)) - \mathbb{P}_1\widehat{V}_2^\pi(s_1,\pi(s_1);r)\\
&\quad + \mathbb{P}_1\widehat{V}_2^\pi(s_1,\pi(s_1);r) + b_1(s_1,\pi(s_1)) - \mathbb{P}_1V_2^\pi(s_1,\pi(s_1);r),1\Big\}\Big]\\
&= \mathbb{E}_{s_1\sim\mu}\Big[\min\Big\{\widehat{f}_1(s_1,\pi(s_1)) - \mathcal{T}_1\widehat{V}_2^\pi(s_1,\pi(s_1)) + \mathbb{P}_1\widehat{V}_2^\pi(s_1,\pi(s_1);r)\\
&\qquad + b_1(s_1,\pi(s_1)) - \mathbb{P}_1V_2^\pi(s_1,\pi(s_1);r),1\Big\}\Big]\\
&\le \mathbb{E}_{s_1\sim\mu}\Big[\min\Big\{2b_1(s_1,\pi(s_1)) + \mathbb{P}_1\widehat{V}_2^\pi(s_1,\pi(s_1);r) - \mathbb{P}_1V_2^\pi(s_1,\pi(s_1);r),1\Big\}\Big],
\end{aligned}$$

where the last inequality holds due to Lemma A.6. Then, by the induction, we have

$$\begin{aligned}
&\mathbb{E}_{s_1\sim\mu}[\widehat{V}_1(s_1) - V_1^\pi(s_1;r)] \\
&\le \mathbb{E}_{s_1\sim\mu}\Big[\min\Big\{2b_1(s_1,\pi(s_1)) + \mathbb{P}_1\widehat{V}_2^\pi(s_1,\pi(s_1);r) - \mathbb{P}_1 V_2^\pi(s_1,\pi(s_1);r), 1\Big\}\Big] \\
&= \mathbb{E}_{s_1\sim\mu, s_2\sim\mathbb{P}(\cdot|s_1,\pi(s_1))}\Big[\min\Big\{2b_1(s_1,\pi(s_1)) + \widehat{V}_2^\pi(s_2;r) - V_2^\pi(s_2;r), 1\Big\}\Big] \\
&\le \mathbb{E}_{\tau\sim d^\pi}\Big[\min\Big\{\sum_{h=1}^H 2b_h(s_h,\pi(s_h)), 1\Big\}\Big] \\
&\le 2\mathbb{E}_{s_1\sim\mu}\Big[\widetilde{V}_1^\pi(s_1;b)\Big] \\
&\le 2\mathbb{E}_{s_1\sim\mu}\Big[\widetilde{V}_1^*(s_1;b)\Big] \\
&= O\Big(\beta^E\sqrt{H\dim_{\alpha,K}(\mathcal{F})/K}\sqrt{\log N_{\mathcal{F}}(\epsilon)/\log N_{\mathcal{V}}(\epsilon)}\Big).
\end{aligned}$$

Therefore, by substituting $\beta^E = \widetilde{O}\big(\sqrt{H\log N_{\mathcal{V}}(\epsilon)}\big)$, we complete the proof:

$$\mathbb{E}_{s_1\sim\mu}[V_1^*(s_1;r) - V_1^\pi(s_1;r)] = O\Big(H\sqrt{\dim_{\alpha,K}(\mathcal{F})/K}\sqrt{\log N_{\mathcal{F}}(\epsilon)}\Big).$$

□

Proof of Corollary 5.2. By solving $\mathbb{E}_{s_1\sim\mu}[V_1^*(s_1;r) - V_1^\pi(s_1;r)] \le \epsilon$, we have that

$$K \ge \frac{H^2\log N_{\mathcal{F}}(\epsilon)\dim_{\alpha,K}(\mathcal{F})}{\epsilon^2}.$$

□

B Proof of Lemmas in Appendix A

In this section, we prove the lemmas used in Appendix A.

Proof of Lemma A.2. We first prove that $\mathcal{E}_{k,h}^E$ holds with probability $1-\delta/(KH)$. We have $\mathcal{T}_h V_{k,h+1} \in \mathcal{F}_h$ due to Assumption 3.3. For any function $V: S\to[0,1]$, let $\eta_h^k(V) = r_{k,h}(s_h^k,a_h^k) + V(s_{h+1}^k) - \mathcal{T}_h V(s_h^k,a_h^k)$. For all $f\in\mathcal{F}_h$, since $a^2 - 2ab = (a-b)^2 - b^2$, we have

$$\begin{aligned}
&\sum_{i\in[k-1]}\frac{1}{(\bar\sigma_{i,h})^2}\Big(f(s_h^i,a_h^i) - \mathcal{T}_h V_{k,h+1}(s_h^i,a_h^i)\Big)^2 - 2\underbrace{\sum_{i\in[k-1]}\frac{1}{(\bar\sigma_{i,h})^2}\Big(f(s_h^i,a_h^i) - \mathcal{T}_h V_{k,h+1}(s_h^i,a_h^i)\Big)\eta_h^k(V_{k,h+1})}_{I(f,\mathcal{T}_h V_{k,h+1},V_{k,h+1})} \\
&= \sum_{i\in[k-1]}\frac{1}{(\bar\sigma_{i,h})^2}\Big(r_{k,h}(s_h^i,a_h^i) + V_{k,h+1}(s_{h+1}^i) - f(s_h^i,a_h^i)\Big)^2 - \sum_{i\in[k-1]}\frac{1}{(\bar\sigma_{i,h})^2}\eta_h^k(V_{k,h+1})^2.
\end{aligned}$$

Take $f = \widehat{f}_{k,h}$. By the the definition of $\widehat{f}_{k,h}$, we have

$$\sum_{i\in[k-1]}\frac{1}{(\bar\sigma_{i,h})^2}\Big(\widehat{f}_{k,h}(s_h^i,a_h^i) - \mathcal{T}_h V_{k,h+1}(s_h^i,a_h^i)\Big)^2 - 2I(\widehat{f}_{k,h},\mathcal{T}_h V_{k,h+1},V_{k,h+1}) \le 0$$

Applying Lemma E.1, for fixed f, $\bar f$, and V, with probability at least $1-\delta$,

$$\begin{aligned}
I(f,\bar f,V) &:= \sum_{i\in[k-1]}\frac{1}{(\bar\sigma_{i,h})^2}\Big(f(s_h^i,a_h^i) - \bar f(s_h^i,a_h^i)\Big)\eta_h^k(V) \\
&\le \frac{2\tau}{\alpha^2}\sum_{i\in[k-1]}\frac{1}{(\bar\sigma_{i,h})^2}\Big(f(s_h^i,a_h^i) - \bar f(s_h^i,a_h^i)\Big)^2 + \frac{1}{\tau}\cdot\log\frac{1}{\delta}.
\end{aligned}$$

Applying a union bound and take $\tau = \frac{\alpha^2}{8}$, for any k, with probability at least $1-\delta$, we have for all V^c in the ϵ-net $\mathcal{V}_{h+1}$ that

$$I(f, \bar{f}, V^c) \leq \frac{1}{4} \sum_{i \in [k-1]} \frac{1}{(\bar{\sigma}_{i,h})^2} \Big(f(s_h^i, a_h^i) - \bar{f}(s_h^i, a_h^i) \Big)^2 + \frac{2}{\alpha^2} \cdot \log \frac{N_{\mathcal{V}}(\epsilon)}{\delta}$$

For all V such that $\|V - V^c\|_\infty \leq \epsilon$, we have $|\eta_h^i(V) - \eta_h^i(V^c)| \leq 4\epsilon$. Thus,

$$I(f, \bar{f}, V_{k,h+1}) \leq \frac{1}{4} \sum_{i \in [k-1]} \frac{1}{(\bar{\sigma}_{i,h})^2} \Big(f(s_h^i, a_h^i) - \bar{f}(s_h^i, a_h^i) \Big)^2 + \frac{2}{\alpha^2} \cdot \log \frac{N_{\mathcal{V}}(\epsilon)}{\delta} + 4\epsilon \cdot kL/\alpha^2$$

Applying a union bound, for any k, with probability at least $1-\delta$, we have for all f^a, f^b in the ϵ-net $\mathcal{C}(\mathcal{F}_h, \epsilon)$ that

$$I(f^a, f^b, V_{k,h+1}) \leq \frac{1}{4} \sum_{i \in [k-1]} \frac{1}{(\bar{\sigma}_{i,h})^2} \Big(f^a(s_h^i, a_h^i) - f^b(s_h^i, a_h^i) \Big)^2 + \frac{2}{\alpha^2} \cdot \log \frac{N_{\mathcal{V}}(\epsilon) \cdot N_{\mathcal{F}}(\epsilon)^2}{\delta} + 4\epsilon \cdot kL/\alpha^2.$$

Therefore, with probability at least $1-\delta$, we have

$$\begin{aligned}
&I(\widehat{f}_{k,h}, \mathcal{T}_h V_{k,h+1}, V_{k,h+1}) \leq I(f^a, f^b, V_{k,h+1}) + 8\epsilon \cdot k/\alpha^2 \\
&\leq \frac{1}{4} \sum_{i \in [k-1]} \frac{1}{(\bar{\sigma}_{i,h})^2} \Big(f^a(s_h^i, a_h^i) - f^b(s_h^i, a_h^i) \Big)^2 + \frac{4}{\alpha^2} \cdot \log \frac{N_{\mathcal{V}}(\epsilon) \cdot N_{\mathcal{F}}(\epsilon)}{\delta} + 4\epsilon \cdot kL/\alpha^2 + 8\epsilon \cdot k/\alpha^2 \\
&\leq \frac{1}{4} \sum_{i \in [k-1]} \frac{1}{(\bar{\sigma}_{i,h})^2} \Big(\widehat{f}_{k,h}(s_h^i, a_h^i) - \mathcal{T}_h V_{k,h+1}(s_h^i, a_h^i) \Big)^2 + \frac{4}{\alpha^2} \cdot \log \frac{N_{\mathcal{V}}(\epsilon) \cdot N_{\mathcal{F}}(\epsilon)}{\delta} + 4\epsilon \cdot kL/\alpha^2 \\
&\quad + 8\epsilon \cdot k/\alpha^2 + 2L\epsilon \cdot k/\alpha^2 \\
&\leq \frac{1}{4} \sum_{i \in [k-1]} \frac{1}{(\bar{\sigma}_{i,h})^2} \Big(\widehat{f}_{k,h}(s_h^i, a_h^i) - \mathcal{T}_h V_{k,h+1}(s_h^i, a_h^i) \Big)^2 + \frac{4}{\alpha^2} \cdot \log \frac{N_{\mathcal{V}}(\epsilon) \cdot N_{\mathcal{F}}(\epsilon)}{\delta} + 14L\epsilon \cdot k/\alpha^2.
\end{aligned}$$

Substituting it back, with probability at least $1 - \delta/(KH)$, we have

$$\frac{1}{4} \sum_{i \in [k-1]} \frac{1}{(\bar{\sigma}_{i,h})^2} \Big(\widehat{f}_{k,h}(s_h^i, a_h^i) - \mathcal{T}_h V_{k,h+1}(s_h^i, a_h^i) \Big)^2 \leq \frac{16}{\alpha^2} \cdot \log \frac{KH \cdot N_{\mathcal{V}}(\epsilon) \cdot N_{\mathcal{F}}(\epsilon)}{\delta} + 56L\epsilon \cdot k/\alpha^2$$

Take $\alpha = 1/\sqrt{H}$ and let

$$\beta^E = \sqrt{16H \log \frac{KH \cdot N_{\mathcal{V}}(\epsilon) \cdot N_{\mathcal{F}}(\epsilon)}{\delta} + 56L\epsilon \cdot K/\alpha^2 + \lambda}.$$

Then we complete the proof by taking a union bound for all $k \in [K]$ and $h \in [H]$. □

Proof of Lemma A.3. We have $\mathcal{T}_h \widehat{V}_{h+1} \in \mathcal{F}_h$ due to Assumption 3.3. For any function $V : S \to [0,1]$, let $\eta_h^k(V) = r_h(s_h^k, a_h^k) + V(s_{h+1}^k) - \mathcal{T}_h V(s_h^k, a_h^k)$. For all $f \in \mathcal{F}_h$, we have

$$\begin{aligned}
&\sum_{i \in [K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2} \Big(f(s_h^i, a_h^i) - \mathcal{T}_h \widehat{V}_{h+1}(s_h^i, a_h^i) \Big)^2 - 2 \underbrace{\sum_{i \in [K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2} \Big(f(s_h^i, a_h^i) - \mathcal{T}_h \widehat{V}_{h+1}(s_h^i, a_h^i) \Big) \eta_h^k(\widehat{V}_{h+1})}_{I(f, \mathcal{T}_h \widehat{V}_{h+1}, \widehat{V}_{h+1})} \\
&= \sum_{i \in [K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2} \Big(r_h(s_h^i, a_h^i) + \widehat{V}_{h+1}(s_{h+1}^i) - f(s_h^i, a_h^i) \Big)^2 - \sum_{i \in [K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2} \eta_h^k(\widehat{V}_{h+1})^2.
\end{aligned}$$

By definition, we have that

$$\sum_{i \in [K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2} \Big(\widehat{f}_h(s_h^i, a_h^i) - \mathcal{T}_h \widehat{V}_{h+1}(s_h^i, a_h^i) \Big)^2 - 2I(\widehat{f}_h, \mathcal{T}_h \widehat{V}_{h+1}, \widehat{V}_{h+1}) \leq 0.$$

We decompose $I(\widehat{f}_h, \mathcal{T}_h\widehat{V}_{h+1}, \widehat{V}_{h+1})$ into two parts:

$$I(\widehat{f}_h, \mathcal{T}_h\widehat{V}_{h+1}, \widehat{V}_{h+1}) = \sum_{i\in[K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(\widehat{f}_h(s_h^i, a_h^i) - \mathcal{T}_h\widehat{V}_{h+1}(s_h^i, a_h^i)\Big)\eta_h^k(\widehat{V}_{h+1} - V_{h+1}^*) + \sum_{i\in[K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(\widehat{f}_h(s_h^i, a_h^i) - \mathcal{T}_h\widehat{V}_{h+1}(s_h^i, a_h^i)\Big)\eta_h^k(V_{h+1}^*). \tag{B.1}$$

For the first term in (B.1), we have

$$\mathbb{E}\Big[\frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(f(s_h^i, a_h^i) - \bar{f}(s_h^i, a_h^i)\Big)\eta_h^k(\widehat{V}_{h+1} - V_{h+1}^*)\Big] = 0.$$

Furthermore, we can bound the maximum as following:

$$\begin{aligned}
&\max_{i\in[K]}\Big|\frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(f(s_h^i, a_h^i) - \bar{f}(s_h^i, a_h^i)\Big)\eta_h^k(\widehat{V}_{h+1} - V_{h+1}^*)\Big| \\
&\le 2\max_{i\in[K]}\Big|\frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(f(s_h^i, a_h^i) - \bar{f}(s_h^i, a_h^i)\Big)\Big| \\
&\le 2\max_{i\in[K]}\frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\sqrt{D^2_{\mathcal{F}_h}(z_{i,h}; z_{[i-1],h}, \bar{\sigma}_{[i-1],h})\Big(\sum_{s\in[i-1]}\frac{1}{(\bar{\sigma}_{s,h})^2}(f(s_h^s, a_h^s) - \bar{f}(s_h^s, a_h^s))^2 + \lambda\Big)} \\
&\le 2\max_{i\in[K]}\frac{1}{(\bar{\sigma}_{i,h})^2}\sqrt{D^2_{\mathcal{F}_h}(z_{i,h}; z_{[i-1],h}, \bar{\sigma}_{[i-1],h})\Big(\sum_{s\in[i-1]}\frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{s,h})^2}(f(s_h^s, a_h^s) - \bar{f}(s_h^s, a_h^s))^2 + \lambda\Big)} \\
&\le 2\cdot\gamma^{-2}\sqrt{\sum_{s\in[K]}\frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{s,h})^2}(f(s_h^s, a_h^s) - \bar{f}(s_h^s, a_h^s))^2 + \lambda},
\end{aligned}$$

where the first inequality is due to bounded total rewards assumption, the second inequality holds due to Definition 3.4, and the last inequality holds due to Line 1 in Algorithm 1 and Definition 3.7.
We further define $\text{var}(V - V_{h+1}^*)$ as

$$\begin{aligned}
\text{var}(V - V_{h+1}^*) &:= \sum_{i\in[K]}\mathbb{E}\Big[\frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^4}\Big(f(s_h^i, a_h^i) - \bar{f}(s_h^i, a_h^i)\Big)^2\eta_h^k(\widehat{V}_{h+1} - V_{h+1}^*)^2\Big] \\
&\le L^2K/\alpha^4.
\end{aligned}$$

By the definition of the indicator function, we have

$$\text{var}(V - V_{h+1}^*) \le \frac{4}{\eta}\sum_{i\in[K]}\frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(f(s_h^i, a_h^i) - \bar{f}(s_h^i, a_h^i)\Big)^2$$

For fixed f, $\bar{f}$, by applying Lemma E.2 with $V^2 = L^2K/\alpha^4, M = 2L/\alpha^2, \upsilon = \eta^{-1/2}, m = \upsilon^2$, and probability at least $1 - \delta/(N_{\mathcal{F}}(\epsilon)^2 N_{\mathcal{V}}(\epsilon)H)$ we have

$$\begin{aligned}
&\sum_{i\in[K]}\frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(f(s_h^i, a_h^i) - \bar{f}(s_h^i, a_h^i)\Big)\eta_h^k(V - V_{h+1}^*) \\
&\le \iota\sqrt{2\left(2\text{var}(V - V_{h+1}^*) + \eta^{-1}\right)} \\
&+ \frac{2}{3}\iota^2\Big(4\gamma^{-2}\sqrt{\sum_{i\in[K]}\frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\left(f(s_h^i, a_h^i) - \bar{f}(s_h^i, a_h^i)\right)^2 + \lambda} + \eta^{-1}\Big),
\end{aligned}$$

where

$$\iota^2(k,h,\delta) := \log \frac{N_{\mathcal{F}}(\epsilon)^2 \cdot N_{\mathcal{V}}(\epsilon) \cdot (\log(L^2 K\eta/\alpha^4)+2) \cdot (\log(2L\eta/\alpha^2)+2)}{\delta/H}$$

Using a union bound over all $(f, \bar{f}, V) \in \mathcal{C}(\mathcal{F}_h, \epsilon) \times \mathcal{C}(\mathcal{F}_h, \epsilon) \times \mathcal{C}(\mathcal{V}_{h=1}, \epsilon)$, we have the inequality above holds for all such $f, \bar{f}, V$ with probability at least $1-\delta/H$. There exist a V^c_{h+1} in the ϵ-net such that $\|\widehat{V}_{h+1} - V^c_{h+1}\| \le \epsilon$. Then we have

$$\begin{aligned}
&\sum_{i\in[K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(\widehat{f}_h(s^i_h,a^i_h) - \mathcal{T}_h\widehat{V}_{h+1}(s^i_h,a^i_h)\Big)\eta^k_h(\widehat{V}_{h+1} - V^*_{h+1}) \\
&\le O\Big(\iota(k,h,\delta)\eta^{-1/2} + \iota(k,h,\delta)^2\gamma^{-2}\Big) \cdot \sqrt{\sum_{\tau\in[K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{\tau,h})^2}(\widehat{f}_h(s^\tau_h,a^\tau_h) - \mathcal{T}_h V_{h+1}(s^\tau_h,a^\tau_h))^2 + \lambda} \\
&+ O(\epsilon k L/\alpha^2) + O(\iota^2(k,h,\delta)\eta^{-1}) + O(\iota(k,h,\delta)\eta^{-1/2}).
\end{aligned} \tag{B.2}$$

For the second term in (B.1), applying Lemma E.1, for fixed f, $\bar{f}$, and V^*_{h+1}, with probability at least $1-\delta$, we have

$$\begin{aligned}
&\sum_{i\in[K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(f(s^i_h,a^i_h) - \bar{f}(s^i_h,a^i_h)\Big)\eta^k_h(V^*_{h+1}) \\
&\le \frac{1}{4}\sum_{i\in[K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(f(s^i_h,a^i_h) - \bar{f}(s^i_h,a^i_h)\Big)^2 + \frac{8}{\alpha^2}\cdot\log\frac{1}{\delta}.
\end{aligned}$$

Applying a union bound, for any k, with probability at least $1-\delta$, we have for all f^a, f^b in the ϵ-net $\mathcal{F}_h$

$$I(f^a, f^b, V^*_{h+1}) \le \frac{1}{4}\sum_{i\in[k-1]} \frac{\widehat{\mathbb{1}}_{i,h}}{(\bar{\sigma}_{i,h})^2}\Big(f^a(s^i_h,a^i_h) - f^b(s^i_h,a^i_h)\Big)^2 + \frac{8}{\alpha^2}\cdot\log\frac{N_{\mathcal{F}}(\epsilon)^2}{\delta}.$$

Therefore, with probability at least $1-\delta$, we have

$$\begin{aligned}
&I(\widehat{f}_h, \mathcal{T}_h V_{h+1}, V^*_{h+1}) \le I(f^a, f^b, V^*_{h+1}) + 8\epsilon\cdot K/\alpha^2 \\
&\le \frac{1}{4}\sum_{i\in[K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(f^a(s^i_h,a^i_h) - f^b(s^i_h,a^i_h)\Big)^2 + \frac{8}{\alpha^2}\cdot\log\frac{\cdot N_{\mathcal{F}}(\epsilon)^2}{\delta} + 8\epsilon\cdot k/\alpha^2 \\
&\le \frac{1}{4}\sum_{i\in[K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(\widehat{f}_h(s^i_h,a^i_h) - \mathcal{T}_h V_{h+1}(s^i_h,a^i_h)\Big)^2 + \frac{8}{\alpha^2}\cdot\log\frac{N_{\mathcal{F}}(\epsilon)^2}{\delta} \\
&\quad + 8\epsilon\cdot k/\alpha^2 + 2L\epsilon\cdot k/\alpha^2 \\
&\le \frac{1}{4}\sum_{i\in[K]} \frac{\widehat{\mathbb{1}}_h}{(\bar{\sigma}_{i,h})^2}\Big(\widehat{f}_h(s^i_h,a^i_h) - \mathcal{T}_h V_{h+1}(s^i_h,a^i_h)\Big)^2 + \frac{8}{\alpha^2}\cdot\log\frac{N_{\mathcal{F}}(\epsilon)^2}{\delta} + 10L\epsilon\cdot k/\alpha^2.
\end{aligned} \tag{B.3}$$

Taking $\eta = \log N_{\mathcal{V}}(\epsilon)$, $\gamma = \widetilde{O}\big(\sqrt{\log N_{\mathcal{V}}(\epsilon)}\big)$ and $\alpha = 1/\sqrt{H}$ and substituting (B.2) and (B.3) back into (B.1), we have

$$\begin{aligned}
&\lambda + \sum_{i\in[K]} \frac{\mathbb{1}_h}{\bar{\sigma}^2_{i,h}}\Big(\widehat{f}_h(s^i_h,a^i_h) - \mathcal{T}_h\widehat{V}_{h+1}(s^i_h,a^i_h)\Big)^2 \\
&\le O\Big(H\log N_{\mathcal{F}}(\epsilon)\Big) + O\Big((\log N_{\mathcal{V}}(\epsilon))^{-1}\log\frac{(\log(L^2K/\alpha^4)+2)\cdot(\log(2L/\alpha^2)+2)}{\delta/H}\Big) + O(\lambda).
\end{aligned}$$

□

Lemma B.1. *On the event $\mathcal{E}^E \cap \mathcal{E}^P_h$, for any $h \in [H]$, we have*

$$V^*_h(s;r) + V_{k,h}(s) \ge \widehat{V}_h(s). \tag{B.4}$$

Lemma B.2. *On the event $\mathcal{E}^E \cap \mathcal{E}^P_{h+1}$, for each episode $k \in [K]$, we have*

$$\log N_{\mathcal{V}}(\epsilon) \cdot [\mathbb{V}_h(\widehat{V}_{h+1} - V^*_{h+1})](s^k_h, a^k_h) \le \sigma^2_{k,h},$$

where $\sigma^2_{k,h} = 4 \log N_{\mathcal{V}}(\epsilon) \cdot \min\{\widehat{f}_{k,h}(s^k_h, a^k_h), 1\}$.

Proof of Lemma A.4. Recall that the indicator function in event $\overline{\mathcal{E}}^P$ is

$$\begin{aligned}\widehat{\mathbb{1}}_h = &\underbrace{\mathbb{1}(V^*_{h+1}(s) \le \widehat{V}_{h+1}(s),\ \forall s \in \mathcal{S})}_{I_1} \cdot \underbrace{\mathbb{1}(\widehat{V}_{h+1}(s) \le V_{k,h+1}(s) + V^*(s;r),\ \forall s \in \mathcal{S}, \forall k \in [K])}_{I_2} \\ &\cdot \underbrace{\mathbb{1}([\mathbb{V}_h(\widehat{V}_{h+1} - V^*_{h+1})](s^k_h, a^k_h) \le \eta^{-1}\bar{\sigma}^2_{k,h},\ \forall k \in [K])}_{I_3},\end{aligned}$$

where $\eta = \log N_{\mathcal{V}}(\epsilon)$. Lemma B.1, Lemma A.7, and Lemma B.2 indicate that $I_1 = I_2 = I_3 = 1$. □

Proof of Lemma A.5. There exists an $\epsilon/2$-net of $\mathcal{F}$, denoted by $\mathcal{C}(\mathcal{F}_h, \epsilon/2)$, such that for any $f \in \mathcal{F}_h$, we can find $f' \in \mathcal{C}(\mathcal{F}, \epsilon/2)$ such that $\|f - f'\|_\infty \le \epsilon/2$. Also, there exists an $\epsilon/2\beta$-net of $\mathcal{B}$, $\mathcal{C}(\mathcal{B}, \epsilon/2\beta)$.
Then we consider the following subset of $\mathcal{V}_h$,

$$\mathcal{V}^c_h = \left\{ V(\cdot) = \max_{a \in \mathcal{A}} \min\left(1, f(\cdot, a) + \beta \cdot b(\cdot, a)\right) \middle| f \in \mathcal{C}(\mathcal{F}_h, \epsilon/2), b \in \mathcal{C}(\mathcal{B}, \epsilon/2\beta) \right\}.$$

Consider an arbitrary $V \in \mathcal{V}$ where $V = \max_{a \in \mathcal{A}} \min(1, f_i(\cdot, a) + \beta \cdot b_i(\cdot, a))$. For each f_i, there exists $f^c_i \in \mathcal{C}(\mathcal{F}_h, \epsilon/2)$ such that $\|f_i - f^c_i\|_\infty \le \epsilon/2$. There also exists $b^c \in \mathcal{C}(\mathcal{B}, \epsilon/2\beta)$ such that $\|b_i - b^c\|_\infty \le \epsilon/2\beta$. Let $V^c = \max_{a \in \mathcal{A}} \min(1, f^c_i(\cdot, a) + \beta \cdot b^c(\cdot, a)) \in \mathcal{V}^c$. It is then straightforward to check that $\|V - V^c\|_\infty \le \epsilon/2 + \beta \cdot \epsilon/2\beta = \epsilon$. By direct calculation, we have $|\mathcal{V}^c_h| = N_{\mathcal{F}_h}(\epsilon/2) \cdot N_{\mathcal{B}}(\epsilon/2\beta)$. □

Proof of Lemma A.6. According to the definition of $D^2_{\mathcal{F}}$ function, we have

$$\begin{aligned}&\left(\widehat{f}_{k,h}(s,a) - \mathcal{T}_h V_{k,h+1}(s,a)\right)^2 \\ &\le D^2_{\mathcal{F}_h}(z; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}) \times \left(\lambda + \sum_{i=1}^{k-1} \frac{1}{(\bar{\sigma}_{i,h})^2} \left(\widehat{f}_{k,h}(s^i_h, a^i_h) - \mathcal{T}_h V_{k,h+1}(s^i_h, a^i_h)\right)^2\right) \\ &\le (\beta^E)^2 \times D^2_{\mathcal{F}_h}(z; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}),\end{aligned}$$

where the first inequality holds due the definition of $D^2_{\mathcal{F}}$ function with the Assumption 3.3 and the second inequality holds due to the events $\mathcal{E}^E_h$. Thus, we have

$$\left|\widehat{f}_{k,h}(s,a) - \mathcal{T}_h V_{k,h+1}(s,a)\right| \le \beta^E D_{\mathcal{F}_h}(z; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}).$$

□

Proof of Lemma A.7. We prove this statement by induction. Note that $V^*_{H+1}(s;r) = \widehat{V}_{H+1}(s)$. Assume that the statement holds for $h+1$. If $\widehat{V}_h(s) = 1$, then the statement holds trivially for h; otherwise, we have for any $(s,a) \in \mathcal{S} \times \mathcal{A}$ that

$$\begin{aligned}&\widehat{Q}_h(s,a) - Q^*_h(s,a;r) \\ &= \widehat{f}_h(s,a) + b_h(s,a) - [r_h(s,a;r) + \mathbb{P}_h V^*_{h+1}(s,a;r)] \\ &= [\widehat{f}_h(s,a) - r_h(s,a;r) - \mathbb{P}_h \widehat{V}_{h+1}(s,a;r)] + b_h(s,a) + \mathbb{P}_h \widehat{V}_{h+1}(s,a;r) - \mathbb{P}_h V^*_{h+1}(s,a;r) \\ &\ge [\widehat{f}_h(s,a) - r_h(s,a;r) - \mathbb{P}_h \widehat{V}_{h+1}(s,a;r)] + b_h(s,a) \\ &\ge -\beta^P D_{\mathcal{F}_h}(z; z_{[K],h}, \bar{\sigma}_{[K],h}) + \beta^P \overline{D}_{\mathcal{F}_h}(z; z_{[K],h}, \bar{\sigma}_{[K],h}) \\ &\ge 0,\end{aligned}$$

where the first inequality holds due to the induction assumption, and the second inequality holds due to Lemma A.6. □

In order to prove Lemma A.8, we need the following three lemmas.

Lemma B.3 (Simulation Lemma)**.** *On the event $\underline{\mathcal{E}}^E$, we have*

$$0 \le V_{k,h}(s_h^k) \le \mathbb{E}_{\tau_h^k \sim d_h^{\pi^k}(s_h^k)} \min\Big\{3\beta^E \sum_{h'=h}^{H} \overline{\mathcal{D}}(z_{k,h'}; z_{[k-1],h'}, \overline{\sigma}_{[k-1],h'}), 1\Big\}.$$

Lemma B.4. *[Lemma C.13 in Zhao et al. (2023)]For any parameters $\beta \ge 1$ and stage $h \in [H]$, the summation of confidence radius over episode $k \in [K]$ is upper bounded by*

$$\begin{aligned}
&\sum_{k=1}^{K} \min\Big(\beta D_{\mathcal{F}_h}(z; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}), 1\Big) \\
&\le (1 + C\beta\gamma^2)\dim_{\alpha,K}(\mathcal{F}_h) + 2\beta\sqrt{\dim_{\alpha,K}(\mathcal{F}_h)}\sqrt{\sum_{k=1}^{K}(\sigma_{k,h}^2 + \alpha^2)},
\end{aligned}$$

where $z = (s, a)$ and $z_{[k-1],h} = \{z_{1,h}, z_{2,h}, .., z_{k-1,h}\}$.

Lemma B.5. *Under event $\underline{\mathcal{E}}^E$, we have*

$$\begin{aligned}
\sum_{k=1}^{K}\sum_{h=1}^{H} \sigma_{k,h}^2 &\le 2304C^2H^3(\log N_{\mathcal{V}}(\epsilon))^2(\beta^E)^2 \dim_{\alpha,K}(\mathcal{F}) \\
&\quad + 48H^2 \log N_{\mathcal{V}}(\epsilon)(1 + C\beta^E\gamma^2)\dim_{\alpha,K}(\mathcal{F}_h) + 16H\log N_{\mathcal{V}}(\epsilon)\sqrt{2HK\log(H/\delta)} + K.
\end{aligned}$$

Now we can prove Lemma A.8.

Proof of Lemma A.8. We have

$$\begin{aligned}
\sum_{k=1}^{K} V_{k,1}(s_1^k) &\le \sum_{k=1}^{K} \mathbb{E}_{\tau_h^k \sim d_h^{\pi^k}(s_h^k)} \min\Big\{3\beta^E \sum_{h'=1}^{H} \overline{\mathcal{D}}(z_{k,h}; z_{[k],h}, \overline{\sigma}_{[k],h}), 1\Big\} \\
&\le \sum_{k=1}^{K}\sum_{h=1}^{H} \mathbb{E}_{\tau_h^k \sim d_h^{\pi^k}(s_h^k)} \min\Big\{3\beta^E \overline{\mathcal{D}}(z_{k,h}; z_{[k-1],h}, \overline{\sigma}_{[k-1],h}), 1\Big\} \\
&\le H(1 + 4C\beta^E\gamma^2)\dim_{\alpha,K}(\mathcal{F}_h) + 8\beta^E\sqrt{\dim_{\alpha,K}(\mathcal{F})}\sqrt{H\sum_{k=1}^{K}\sum_{h=1}^{H}(\sigma_{k,h}^2 + \alpha^2)} \\
&= O(\beta^E\sqrt{KH\dim_{\alpha,K}(\mathcal{F})}),
\end{aligned}$$

where the first inequality follows from Lemma B.3, the third inequality follows from Lemma B.4, and the last equality holds due to Lemma B.5. □

Proof of Lemma A.9. Denote $\Delta_k = \mathbb{E}_{s\sim\mu}\big[\widetilde{V}_1^*(s; r_k)\big] - \widetilde{V}_1^*(s_1^k; r_k)$. By Azuma-Hoeffding inequality (Lemma E.3), we have

$$\Big|\sum_{k=1}^{K} \Delta_k\Big| \le \sqrt{2K\log(1/\delta)}.$$

□

Lemma B.6. *On the event $\underline{\mathcal{E}}^E$, for any $k \in [K]$ and $h \in [H]$, we have*

$$\widetilde{V}_h^*(s; r_k) \le V_{k,h}(s), \quad \forall s \in \mathcal{S}.$$

Proof of Lemma A.10. Since $\beta^E = O(\sqrt{H \log N_{\mathcal{V}}(\epsilon)})$ and $\beta^P = O(\sqrt{H \log N_{\mathcal{F}}(\epsilon)})$, for some constant c, we have

$$\beta^E \geq c\sqrt{\log N_{\mathcal{V}}(\epsilon)/\log N_{\mathcal{F}}(\epsilon)} \cdot \beta^P.$$

Therefore, for any $h \in [H]$, we have $r_{k,h}(\cdot,\cdot) \geq r_{K,h}(\cdot,\cdot) \geq c\sqrt{\log N_{\mathcal{V}}(\epsilon)/\log N_{\mathcal{F}}(\epsilon)} \cdot b_h(\cdot,\cdot)$. Hence,

$$\begin{aligned}
&c\sqrt{\log N_{\mathcal{V}}(\epsilon)/\log N_{\mathcal{F}}(\epsilon)} \cdot \mathbb{E}_{s\sim\mu}\Big[\widetilde{V}_1^*(s;b)\Big] \\
&= \mathbb{E}_{s\sim\mu}\Big[\widetilde{V}_1^*(s; c\sqrt{\log N_{\mathcal{V}}(\epsilon)/\log N_{\mathcal{F}}(\epsilon)} \cdot b)\Big] \\
&\leq \mathbb{E}_{s\sim\mu}\Big[\widetilde{V}_1^*(s;r_k)\Big]/K \\
&= \Big[\sum_{k=1}^K \widetilde{V}_1^*(s_1^k;r_k) + \sum_{k=1}^K \Big[\mathbb{E}_{s\sim\mu}\Big[\widetilde{V}_1^*(s;r_k)\Big] - \widetilde{V}_1^*(s_1^k;r_k)\Big]\Big]/K \\
&\leq \Big(\sum_{k=1}^K \widetilde{V}_1^*(s;r_k)\Big)/K + \sqrt{2\log(1/\delta)/K} \\
&\leq \Big(\sum_{k=1}^K V_{k,1}(s;r_k)\Big)/K + \sqrt{2\log(1/\delta)/K} \\
&= O\Big(\beta^E\sqrt{H\dim_{\alpha,K}(\mathcal{F})/K}\Big),
\end{aligned}$$

where the second inequality follows from Lemma A.9, and the third inequality follows from Lemma B.6. Therefore, we have

$$\mathbb{E}_{s\sim\mu}\Big[\widetilde{V}_1^*(s;b)\Big] = O\Big(\beta^E\sqrt{H\dim_{\alpha,K}(\mathcal{F})/K}\sqrt{\log N_{\mathcal{F}}(\epsilon)/\log N_{\mathcal{V}}(\epsilon)}\Big).$$

□

C Proofs of Lemmas in Appendix B

Proof of Lemma B.1. We see that

$$\begin{aligned}
Q^*(\cdot,\cdot;r) &= r_h(\cdot,\cdot) + \mathbb{P}_h V_{h+1}(\cdot,\cdot;r), \\
Q_{k,h}(\cdot,\cdot) &= \min\{\widehat{f}_{k,h}(\cdot,\cdot) + b_{k,h}(\cdot,\cdot), 1\}, \\
\widehat{Q}_h(\cdot,\cdot) &= \min\{\widehat{f}_h(\cdot,\cdot) + b_h(\cdot,\cdot), 1\}.
\end{aligned}$$

We prove this statement by induction. Note that $V_{H+1}^*(s;r) + V_{k,H+1}(s) = \widehat{V}_{H+1}(s) = 0$. Assume the statement holds for $h+1$. By definition, we have

$$Q_h^*(s,a;r) + 1 \geq \widehat{Q}_h(s,a).$$

Therefore, we only need to prove

$$Q_h^*(s,a;r) + \widehat{f}_{k,h}(s,a) + b_{k,h}(s,a) - \widehat{Q}_h(s,a) \geq 0.$$

We have

$$
\begin{aligned}
&Q_h^*(s,a;r) + \widehat{f}_{k,h}(s,a) + b_{k,h}(s,a) - \widehat{Q}_h(s,a) \\
&= r_h(s,a) + \mathbb{P}_h V_{h+1}^*(s,a;r) + \widehat{f}_{k,h}(s,a) + b_{k,h}(s,a) - \min\{\widehat{f}_h(s,a) + b_h(s,a), 1\} \\
&\geq r_h(s,a) + \mathbb{P}_h V_{h+1}^*(s,a;r) + \widehat{f}_{k,h}(s,a) + b_{k,h}(s,a) - (\widehat{f}_h(s,a) + b_h(s,a)) \\
&= \mathbb{P}_h V_{h+1}^*(s,a;r) + \mathbb{P}_h V_{k,h+1}(s,a) - \mathbb{P}_h \widehat{V}_{h+1}(s,a) + \widehat{r}_{k,h}(s,a) + b_{k,h}(s,a) - b_h(s,a) \\
&\qquad + (\widehat{f}_{k,h}(s,a) - \widehat{r}_{k,h}(s,a) - \mathbb{P}_h V_{k,h+1}(s,a)) + (r_h(s,a) + \mathbb{P}_h \widehat{V}_h(s,a) - \widehat{f}_h(s,a)) \\
&\geq \widehat{r}_{k,h}(s,a) + b_{k,h}(s,a) - b_h(s,a) + (\widehat{f}_{k,h}(s,a) - \widehat{r}_{k,h}(s,a) - \mathbb{P}_h V_{k,h+1}(s,a)) \\
&\qquad + (r_h(s,a) + \mathbb{P}_h \widehat{V}_h(s,a) - \widehat{f}_h(s,a)) \\
&\geq 3\beta^E \overline{\mathcal{D}}_{\mathcal{F}_h}(z; z_{[k-1],h}, \overline{\sigma}_{[k-1],h}) - \beta^P \overline{\mathcal{D}}_{\mathcal{F}_h}(z; z_{[K],h}, \overline{\sigma}_{[K],h}) - \beta^E \mathcal{D}_{\mathcal{F}_h}(z; z_{[k-1],h}, \overline{\sigma}_{[k-1],h}) \\
&\qquad - \beta^P \mathcal{D}_{\mathcal{F}_h}(z; z_{[K],h}, \overline{\sigma}_{[K],h}) \\
&\geq 0,
\end{aligned}
$$

where the second inequality holds due to induction assumption, the third inequality holds by high probability events, and the last inequality holds by $\beta^E \geq \beta^P$, $\bar{\mathcal{D}}_{\mathcal{F}_h}(z; z_{[k],h}, \overline{\sigma}_{[k],h})$ decreasing with k, and Definition 3.6. □

Lemma C.1. *On the event $\underline{\mathcal{E}}^E$, we have*

$$
|\widehat{f}_{k,h}(s,a) - \mathcal{T}_h V_{k,h+1}| \leq \beta^E D_{\mathcal{F}_h}(z; z_{[k-1],h}, \bar{\sigma}_{[k-1],h})
$$

Proof of Lemma B.2. We have Lemma A.7 and B.1 both hold on $\mathcal{E}_{h+1}^P$. Therefore, we have

$$
\begin{aligned}
&[\mathbb{V}_h(\widehat{V}_{h+1} - V_{h+1}^*)](s_h^k, a_h^k) \\
&\leq [\mathbb{P}_h(\widehat{V}_{h+1} - V_{h+1}^*)^2](s_h^k, a_h^k) \\
&\leq 2[\mathbb{P}_h(\widehat{V}_{h+1} - V_{h+1}^*)](s_h^k, a_h^k) \\
&\leq 2[\mathbb{P}_h V_{k,h+1}](s_h^k, a_h^k) \\
&= 2(\mathcal{T}_h V_{k,h+1}(s_h^k, a_h^k) - r_{k,h}(s_h^k, a_h^k)) \\
&\leq 2(\widehat{f}_{k,h}(s_h^k, a_h^k) + \beta^E D_{\mathcal{F}_h}(z_{k,h}; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}) - \beta^E \overline{D}_{\mathcal{F}_h}(z_{k,h}; z_{[k-1],h}, \bar{\sigma}_{[k-1],h})) \\
&\leq 2\widehat{f}_{k,h}(s_h^k, a_h^k),
\end{aligned}
$$

where the second inequality holds due to Lemma A.7 and $\widehat{V}_{h+1}, V_{h+1}^* \in [0,1]$, the third inequality holds due to Lemma B.1, the fourth inequality holds due to Lemma C.1, and the last inequality holds due to Definition 3.6. □

Proof of Lemma B.3. According to Algorithm 1, we have that

$$
\begin{aligned}
Q_{k,h}(\cdot,\cdot) &= \min\{\widehat{f}_{k,h}(\cdot,\cdot) + b_{k,h}(\cdot,\cdot), 1\}, \\
V_{k,h}(\cdot) &= \max_a Q_{k,h}(\cdot, a), \\
a_h^k &= \pi_h^k(s_h^k) = \operatorname*{argmax}_a Q_{k,h}(s_h^k, a).
\end{aligned}
$$

For all k and all h, we have that

$$\begin{aligned}
V_{k,h}(s_h^k) &= Q_{k,h}(s_h^k, a_h^k) \\
&\leq \widehat{f}_{k,h}(s_h^k, a_h^k) + b_{k,h}(s_h^k, a_h^k) \\
&= 2\beta^E \overline{\mathcal{D}}(z_{k,h}; z_{[k-1],h}, \overline{\sigma}_{[k-1],h}) + (\widehat{f}_{k,h}(s_h^k, a_h^k) - \mathcal{T}_h V_{k,h+1}(s_h^k, a_h^k)) + \mathcal{T}_h V_{k,h+1}(s_h^k, a_h^k) \\
&\cdots \\
&= \mathbb{E}_{\tau_h^k \sim d_h^{\pi^k}(s_h^k)} \sum_{h'=h}^{H} \left[(\widehat{f}_{k,h'}(s_{h'}^k, a_{h'}^k) - \mathcal{T}_h V_{k,h'+1}(s_{h'}^k, a_{h'}^k)) + 2\beta^E \overline{\mathcal{D}}(z_{k,h'}; z_{[k-1],h'}, \overline{\sigma}_{[k-1],h'}) \right] \\
&\leq \mathbb{E}_{\tau_h^k \sim d_h^{\pi^k}(s_h^k)} \sum_{h'=h}^{H} 3\beta^E \overline{\mathcal{D}}(z_{k,h'}; z_{[k-1],h'}, \overline{\sigma}_{[k-1],h'}),
\end{aligned}$$

where the last inequality holds due to Lemma C.1 and Definition 3.6. □

Lemma C.2. *On the event $\underline{\mathcal{E}}^E$, with probability at least $1-\delta$,*

$$\begin{aligned}
\sum_{k=1}^{K} \sum_{h=1}^{H} \mathbb{P}_h V_{k,h+1}(s_h^k, , a_h^k) \leq H \sum_{k=1}^{K} \sum_{h=1}^{H} \min \left\{ 4\beta^E D_{\mathcal{F}_h}(z_{k,h}; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}), 1 \right\} \\
+ (H+1)\sqrt{2HK \log(1/\delta)}
\end{aligned}$$

Proof of Lemma B.5. Recall $\sigma_{k,h}^2 = 4 \log N_{\mathcal{V}}(\epsilon) \cdot \min\{\widehat{f}_{k,h}(s_h^k, a_h^k), 1\}$. We have

$$\begin{aligned}
\sum_{k=1}^{K} \sum_{h=1}^{H} \sigma_{k,h}^2 &= 4 \log N_{\mathcal{V}}(\epsilon) \sum_{k=1}^{K} \sum_{h=1}^{H} \min\{\widehat{f}_{k,h}(s_h^k, a_h^k), 1\} \\
&\leq 4 \log N_{\mathcal{V}}(\epsilon) \sum_{k=1}^{K} \sum_{h=1}^{H} \min\{\mathcal{T}_h V_{k,h+1}(s_h^k, a_h^k) + \beta^E D_{\mathcal{F}_h}(z_{k,h}; z_{[k-1].h}, \bar{\sigma}_{[k-1],h}), 1\} \\
&\leq 4 \log N_{\mathcal{V}}(\epsilon) \sum_{k=1}^{K} \sum_{h=1}^{H} \min\{\mathbb{P}_h V_{k,h+1}(s_h^k, a_h^k) + 2\beta^E \overline{\mathcal{D}}_{\mathcal{F}_h}(z_{k,h}; z_{[k-1].h}, \bar{\sigma}_{[k-1],h}), 1\} \\
&\leq 4 \log N_{\mathcal{V}}(\epsilon) \sum_{k=1}^{K} \sum_{h=1}^{H} \mathbb{P}_h V_{k,h+1}(s_h^k, a_h^k) + 8 \log N_{\mathcal{V}}(\epsilon) \sum_{k=1}^{K} \sum_{h=1}^{H} \{\beta^E \overline{\mathcal{D}}_{\mathcal{F}_h}(z_{k,h}; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}), 1\} \\
&\leq 24H \log N_{\mathcal{V}}(\epsilon) \underbrace{\sum_{k=1}^{K} \sum_{h=1}^{H} \min \left\{ \beta^E \overline{\mathcal{D}}_{\mathcal{F}_h}(z_{k,h}; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}), 1 \right\}}_{I} + 8H \log N_{\mathcal{V}}(\epsilon) \sqrt{2HK \log(H/\delta)},
\end{aligned}$$

where the first inequality holds due to Lemma C.1, the second inequality holds due to Definition 3.6, and the last inequality

holds due to Lemma C.2. For the term I, we further have

$$\begin{aligned}
&\sum_{k=1}^{K}\sum_{h=1}^{H}\min\left\{\beta^E\overline{\mathcal{D}}_{\mathcal{F}_h}(z_{k,h};z_{[k-1],h},\bar{\sigma}_{[k-1],h}),1\right\}\\
&\le\sum_{k=1}^{K}\sum_{h=1}^{H}\min\left\{C\beta^E\mathcal{D}_{\mathcal{F}_h}(z_{k,h};z_{[k-1],h},\bar{\sigma}_{[k-1],h}),1\right\}\\
&\le\sum_{h=1}^{H}(1+C\beta^E\gamma^2)\dim_{\alpha,K}(\mathcal{F}_h)+2C\beta^E\sum_{h=1}^{H}\sqrt{\dim_{\alpha,K}(\mathcal{F}_h)}\sqrt{\sum_{k=1}^{K}(\sigma_{k,h}^2+\alpha^2)}\\
&\le H(1+C\beta^E\gamma^2)\dim_{\alpha,K}(\mathcal{F})+2C\beta^E\sqrt{\sum_{h=1}^{H}\dim_{\alpha,K}(\mathcal{F}_h)}\sqrt{\sum_{k=1}^{K}\sum_{h=1}^{H}(\sigma_{k,h}^2+\alpha^2)}\\
&\le H(1+C\beta^E\gamma^2)\dim_{\alpha,K}(\mathcal{F}_h)+2C\beta^E\sqrt{\dim_{\alpha,K}(\mathcal{F})}\sqrt{H\sum_{k=1}^{K}\sum_{h=1}^{H}(\sigma_{k,h}^2+\alpha^2)},
\end{aligned}$$

where the first inequality holds due to Definition 3.6, the second inequality holds due to Lemma B.4, the third inequality holds due to Cauchy-Schwarz inequality. Therefore, we can get

$$\begin{aligned}
\sum_{k=1}^{K}\sum_{h=1}^{H}\sigma_{k,h}^2 &\le 24H^2\log N_{\mathcal{V}}(\epsilon)(1+C\beta^E\gamma^2)\dim_{\alpha,K}(\mathcal{F}_h)\\
&\quad+48CH\log N_{\mathcal{V}}(\epsilon)\beta^E\sqrt{\dim_{\alpha,K}(\mathcal{F})}\sqrt{H\sum_{k=1}^{K}\sum_{h=1}^{H}(\sigma_{k,h}^2+\alpha^2)}+8H\log N_{\mathcal{V}}(\epsilon)\sqrt{2HK\log(H/\delta)}.
\end{aligned}$$

Since $x\le a\sqrt{x}+b$ implies $x\le a^2+2b$, taking $\alpha=1/\sqrt{H}$, we have that

$$\begin{aligned}
\sum_{k=1}^{K}\sum_{h=1}^{H}\sigma_{k,h}^2 &\le 2304C^2H^3(\log N_{\mathcal{V}}(\epsilon))^2(\beta^E)^2\dim_{\alpha,K}(\mathcal{F})\\
&\quad+48H^2\log N_{\mathcal{V}}(\epsilon)(1+C\beta^E\gamma^2)\dim_{\alpha,K}(\mathcal{F}_h)+16H\log N_{\mathcal{V}}(\epsilon)\sqrt{2HK\log(H/\delta)}+K.
\end{aligned}$$

□

Proof of Lemma B.6. We prove this statement by induction. Note that $\widetilde{V}_{H+1}^*(s;r_k)=V_{k,H+1}(s)=0$. Assume that the statement holds for $h+1$. If $V_{k,h}(s)=1$, then the statement holds trivially for h; otherwise, we have for any $(s,a)\in\mathcal{S}\times\mathcal{A}$ that

$$\begin{aligned}
&\widehat{Q}_{k,h}(s,a)-\widetilde{Q}_h^*(s,a;r_k)\\
&\ge\widehat{f}_{k,h}(s,a)+b_{k,h}(s,a)-[r_{k,h}(s,a;r)+\mathbb{P}_hV_{h+1}^*(s,a;r)]\\
&=[\widehat{f}_{k,h}(s,a)-r_{k,h}(s,a;r)-\mathbb{P}_hV_{k,h+1}(s,a;r)]+b_{k,h}(s,a)+\mathbb{P}_hV_{k,h+1}(s,a;r)-\mathbb{P}_hV_{h+1}^*(s,a;r)\\
&\ge[\widehat{f}_{k,h}(s,a)-r_{k,h}(s,a;r)-\mathbb{P}_hV_{k,h+1}(s,a;r)]+b_{k,h}(s,a)\\
&\ge-\beta^ED_{\mathcal{F}_h}(z;z_{[K],h},\bar{\sigma}_{[K],h})+2\beta^E\overline{D}_{\mathcal{F}_h}(z;z_{[K],h},\bar{\sigma}_{[K],h})\\
&\ge 0,
\end{aligned}$$

where the first inequality holds due to Definition A.1, the second inequality holds due to induction hypothesis, the third inequality holds due to Lemma C.1, and the forth inequality holds due to Definition 3.6. □

D Proof of Lemmas in Appendix C

Proof of Lemma C.1. According to the definition of $D^2_{\mathcal{F}}$ function, we have

$$\begin{aligned}
&\left(\widehat{f}_{k,h}(s,a) - \mathcal{T}_h V_{k,h+1}(s,a)\right)^2 \\
&\leq D^2_{\mathcal{F}_h}(z; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}) \times \left(\lambda + \sum_{i=1}^{k-1} \frac{1}{(\bar{\sigma}_{i,h})^2} \left(\widehat{f}_{k,h}(s_h^i, a_h^i) - \mathcal{T}_h V_{k,h+1}(s_h^i, a_h^i)\right)^2\right) \\
&\leq (\beta^E)^2 \times D^2_{\mathcal{F}_h}(z; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}),
\end{aligned}$$

where the first inequality holds due the definition of $D^2_{\mathcal{F}}$ function with the Assumption 3.3 and the second inequality holds due to the events $\mathcal{E}_h^E$. Thus, we have

$$\left|\widehat{f}_{k,h}(s,a) - \mathcal{T}_h V_{k,h+1}(s,a)\right| \leq \beta^E D_{\mathcal{F}_h}(z; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}).$$

□

Proof of Lemma C.2. By Lemma E.3, we have

$$\begin{aligned}
\sum_{k=1}^K \sum_{h=1}^H \mathbb{P}_h V_{k,h+1}(s_h^k, a_h^k) &= \sum_{k=1}^K \sum_{h=1}^H V_{k,h+1}(s_{h+1}^k) + \sum_{k=1}^K \sum_{h=1}^H (\mathbb{P}_h V_{k,h+1}(s_h^k, a_h^k) - V_{k,h+1}(s_{h+1}^k)) \\
&\leq \sum_{k=1}^K \sum_{h=1}^H V_{k,h+1}(s_{h+1}^k) + \sqrt{2KH\log(1/\delta)}.
\end{aligned}$$

Then, under event $\underline{\mathcal{E}}^E$, we have

$$\begin{aligned}
V_{k,h}(s_h^k) &= Q_{k,h}(s_h^k, a_h^k) \\
&= \min\{\widehat{f}_{k,h}(s_h^k, a_h^k) + 2\beta^E \overline{\mathcal{D}}_{\mathcal{F}_h}(z_{k,h}; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}), 1\} \\
&\leq \min\{\mathbb{P}_h V_{k,h+1}(s_h^k, a_h^k) + 4\beta^E \overline{\mathcal{D}}_{\mathcal{F}_h}(z_{k,h}; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}), 1\} \\
&= \min\{V_{k,h+1}(s_h^k) + (\mathbb{P}_h V_{k,h+1}(s_h^k, a_h^k) - V_{k,h+1}(s_h^k)) + 4\beta^E \overline{\mathcal{D}}_{\mathcal{F}_h}(z_{k,h}; z_{[k-1],h}, \bar{\sigma}_{[k-1],h}), 1\},
\end{aligned}$$

where the inequality holds due to Lemma C.1 and Definition 3.6. Therefore, for fixed h, we have

$$\begin{aligned}
\sum_{k=1}^K V_{k,h}(s_h^k) &\leq \sum_{k=1}^K \min\Big\{ \sum_{h'=h}^H \big[4\beta^E \overline{\mathcal{D}}_{\mathcal{F}_{h'}}(z_{k,h'}; z_{[k-1],h'}, \bar{\sigma}_{[k-1],h'}) \\
&\qquad + (\mathbb{P}_h V_{k,h'+1}(s_{h'}^k, a_{h'}^k) - V_{k,h'+1}(s_{h'}^k))\big], 1\Big\} \\
&\leq \sum_{k=1}^K \sum_{h'=h}^H \min\Big\{4\beta^E \overline{\mathcal{D}}_{\mathcal{F}_{h'}}(z_{k,h'}; z_{[k-1],h'}, \bar{\sigma}_{[k-1],h'}), 1\Big\} \\
&\qquad + \sum_{k=1}^K \sum_{h'=h}^H (\mathbb{P}_h V_{k,h'+1}(s_{h'}^k, a_{h'}^k) - V_{k,h'+1}(s_{h'}^k)) \\
&\leq \sum_{k=1}^K \sum_{h'=h}^H \min\Big\{4\beta^E \overline{\mathcal{D}}_{\mathcal{F}_{h'}}(z_{k,h'}; z_{[k-1],h'}, \bar{\sigma}_{[k-1],h'}), 1\Big\} + \sqrt{2HK\log(1/\delta)},
\end{aligned}$$

where the first inequality holds due to induction, and the last inequality holds due to Lemma E.3. Hence, by combining the

above two inequalities, we have

$$\begin{aligned}
&\sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{P}_h V_{k,h+1}(s_h^k,a_h^k)\\
&\le \sum_{k=1}^{K}\sum_{h=1}^{H} V_{k,h+1}(s_{h+1}^k)+\sqrt{2KH\log(1/\delta)}\\
&\le H\sum_{k=1}^{K}\sum_{h=1}^{H}\min\left\{4\beta^E\overline{\mathcal{D}}_{\mathcal{F}_h}(z_{k,h};z_{[k-1],h},\bar{\sigma}_{[k-1],h}),1\right\}+(H+1)\sqrt{2HK\log(1/\delta)}.
\end{aligned}$$

□

E Auxilliary Lemmas

Lemma E.1 (Self-normalized bound for scalar-valued martingales)**.** *Consider random variables $(v_n|n\in\mathbb{N})$ adapted to the filtration $(\mathcal{H}_n:\ n=0,1,...)$. Let $\{\eta_i\}_{i=1}^{\infty}$ be a sequence of real-valued random variables which is $\mathcal{H}_{i+1}$-measurable and is conditionally σ-sub-Gaussian. Then for an arbitrarily chosen $\lambda>0$, for any $\delta>0$, with probability at least $1-\delta$, it holds that*

$$\sum_{i=1}^{n}\eta_i v_i \le \frac{\lambda\sigma^2}{2}\cdot\sum_{i=1}^{n}v_i^2+\log(1/\delta)/\lambda \qquad \forall n\in\mathbb{N}.$$

Lemma E.2 (Corollary 2, Agarwal et al. (2022))**.** *Let $M>0, V>v>0$ be constants, and $\{x_i\}_{i\in[t]}$ be stochastic process adapted to a filtration $\{\mathcal{H}_i\}_{i\in[t]}$. Suppose $\mathbb{E}[x_i|\mathcal{H}_{i-1}]=0$, $|x_i|\le M$ and $\sum_{i\in[t]}\mathbb{E}[x_i^2|\mathcal{H}_{i-1}]\le V^2$ almost surely. Then for any $\delta,\epsilon>0$, let $\iota=\sqrt{\log\frac{(2\log(V/v)+2)\cdot(\log(M/m)+2)}{\delta}}$ we have*

$$\mathbb{P}\left(\sum_{i\in[t]}x_i>\iota\sqrt{2\left(2\sum_{i\in[t]}\mathbb{E}[x_i^2|\mathcal{H}_{i-1}]+v^2\right)}+\frac{2}{3}\iota^2\left(2\max_{i\in[t]}|x_i|+m\right)\right)\le\delta.$$

Lemma E.3 (Azuma-Hoeffding Inequality)**.** *Let $\{x_i\}_{i=1}^{n}$ be a martingale difference sequence with respect to a filtration $\{\mathcal{G}_i\}_{i=1}^{n+1}$ such that $|x_i|\le M$ almost surely. That is, x_i is $\mathcal{G}_{i+1}$-measurable and $\mathbb{E}[x_i|\mathcal{G}_i]$ a.s. Then for any $0<\delta<1$, with probability at least $1-\delta$,*

$$\sum_{i=1}^{n}x_i\le M\sqrt{2n\log(1/\delta)}.$$

F Experiment details

F.1 Details of exploration algorithm

We present the practical algorithm in this subsection. We start by introducing the notation $\boldsymbol{\phi}_i$ as the parameter for the i-th Q networks, which is a three-layer MLP with 1024 hidden size, same as other benchmark algorithms implemented in URLB (Laskin et al., 2021). For the ease of presentation, we ignore the Q network as $Q_{\boldsymbol{\phi}_i}$ as Q_i and the target network $Q_{\bar{\boldsymbol{\phi}}_i}$ as $\bar{Q}_i$ when there is no confusion. We initialize the parameters in $\boldsymbol{\phi}_i$ using Kaiming distribution (He et al., 2015).

The algorithm works in the discounted MDP with the discounted factor γ. For each t in training steps, the algorithm updates the $t\%N$-th Q function by taking the gradient descent regarding the loss function

$$\mathcal{L}(\boldsymbol{\phi}_{t\%N})=\sum_{(s,a,s')\in\mathcal{B}}\frac{1}{\sigma^2(s,a)}\left(Q_{t\%N}(s,a)-\left(r_{\text{int}}(s,a)+\gamma Q_{\text{target}}(s,a)+b(s,a)\right)\right)^2, \tag{F.1}$$

where the target Q function is the average of N target Q network, i.e., $Q_{\text{target}}(s,a)=\sum_{i\in[N]}\bar{Q}_i(s,a)/N$, $\mathcal{B}$ is the minibatch randomly sampled from replay buffer $\mathcal{D}$. We encourage the diversity of different Q function by using different batch $\mathcal{B}$ for

updating different Q functions. As the key components of our algorithm, weighted regression $\sigma^2(s,a)$; intrinsic reward $r_{\text{int}}(s,a)$, exploration bonus $b(s,a)$ is calculated based on the variance of the target Q network across $\bar{Q}_i$ instances:

$$\sigma^2(s,a) = \text{Var}[\bar{Q}_i(s,a)]; \qquad r_{\text{int}}(s,a) = (1-\gamma)\sqrt{\text{Var}[\bar{Q}_i(s,a)]}; \qquad b(s,a) = \beta\sqrt{\text{Var}[\bar{Q}_i(s,a)]}, \tag{F.2}$$

where we simply set $\beta = 1$ to align with our theory, the factor $(1-\gamma)$ before the intrinsic reward is because we want to balance the horizon $1/H \approx (1-\gamma)$ in the setting. The reason for choosing the target Q function $\bar{Q}_i$ instead of the updating Q function is to update the intrinsic reward, exploration bonus slower than the update of Q function, therefore give the agent more time to explore the optimal policy for maximizing a certain intrinsic reward $r_{\text{int}}(s,a)$. After updating the parameter $\boldsymbol{\phi}_{t\%N}$, we perform a soft update for the target network as

$$\bar{\boldsymbol{\phi}}_{t\%N} \leftarrow (1-\eta)\bar{\boldsymbol{\phi}}_{t\%N} + \eta\boldsymbol{\phi}_{t\%N}, \tag{F.3}$$

where we follow the setting in URLB to set $\eta = 0.01$. After updating the Q function, the algorithm then updates the actor $\pi_{\boldsymbol{\theta}}(a|s)$ following DDPG in maximizing

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{(s,a,s')\in\mathcal{B}} \sum_{i\in[N]} Q_i(s, \pi_{\boldsymbol{\theta}}(a|s)) \tag{F.4}$$

We summarize the exploration algorithm in Algorithm 2, in particular, we use Adam to optimize the loss function defined by (F.1) and (F.3).

Algorithm 2 `GFA-RFE`- Exploration Phase – Implementation

Require: Number of ensemble N, update speed η, exploration step T, (reward-free) environment `env`,
Require: Action variance σ^2, minibatch size B, exploration bonus β, discount factor γ

For all $i \in [N]$, initialize $\boldsymbol{\phi}_i$, let $\bar{\boldsymbol{\phi}}_i \leftarrow \boldsymbol{\phi}_i$
Initialize policy network $\pi_{\boldsymbol{\theta}}$, replay buffer $\mathcal{D} = \emptyset$
Observe initial state s_1
for $t = 1, \cdots, T$ **do**
  Sample $\zeta \sim$ Unif.$[0,1]$, sample $a_t \sim \Big\{N(\pi(\cdot|s_t), \sigma^2)$ **if** $\zeta \le 1-\epsilon$ **else** Unif.$(\mathcal{A})\Big\}$
  Observe s_{t+1}, let $\mathcal{D} \leftarrow \mathcal{D} \cup (s_t, a_t, s_{t+1})$
  If `env.done`, restart `env` and observe initial state s_{t+1}
  Sample a minibatch $\mathcal{B} = \{(s,a,s')\} \subseteq \mathcal{D}$ with size B
  For each (s,a,s') triplet, calculate $\sigma^2(s,a), r_{\text{int}}(s,a), b(s,a)$ according to (F.2).
  Update Q-network $Q_{t\%N}$ by taking one step minimizing $\mathcal{L}(\boldsymbol{\phi}_{t\%N})$ according to (F.1)
  Update actor $\pi_{\boldsymbol{\theta}}(\cdot|s)$ by taking one step maximizing $\mathcal{L}(\boldsymbol{\theta})$ according to (F.4)
  Update target Q-network following (F.3)
end for

Algorithm 3 `GFA-RFE`- Planning Phase – Implementation (DDPG)

Require: Update speed η, training K, environment `env`, reward function $r(\cdot,\cdot)$
Require: Action variance σ^2, minibatch size B, discount factor γ, offline training data $\mathcal{D}$

Initialize $\boldsymbol{\phi}$, let $\bar{\boldsymbol{\phi}} \leftarrow \boldsymbol{\phi}$
Initialize policy network $\pi_{\boldsymbol{\theta}}$
Update every (s,a,s') in $\mathcal{D}$ to $(s,a,s',r(s,a))$
for $k = 1, \cdots, K$ **do**
  Sample a minibatch $\mathcal{B} = \{(s,a,s',r(s,a))\} \subseteq \mathcal{D}$
  Calculate $\mathcal{L}(\boldsymbol{\phi}) = \sum_{(s,a,s')\in\mathcal{B}} \Big(Q_{\boldsymbol{\phi}}(s,a) - \Big(r(s,a) + \gamma Q_{\text{target}}(s', \pi_{\boldsymbol{\theta}}(s'))\Big)\Big)^2$
  Update Q-network $Q_{t\%N}$ by taking one step minimizing $\mathcal{L}(\boldsymbol{\phi})$
  Calculate actor loss $\mathcal{L}(\boldsymbol{\theta}) = \sum_{(s,a,s')\in\mathcal{B}} Q_{\boldsymbol{\phi}}(s, \pi_{\boldsymbol{\theta}}(a|s))$
  Update actor $\pi_{\boldsymbol{\theta}}(\cdot|s)$ by taking one step maximizing $\mathcal{L}(\boldsymbol{\theta})$
  Update target Q-network by $\bar{\boldsymbol{\phi}} \leftarrow (1-\eta)\bar{\boldsymbol{\phi}} + \eta\boldsymbol{\phi}$
end for

Table 3. The common set of hyper-parameters.

Common hyper-parameter	Value
Replay buffer capacity	10^6
Action repeat	1
n-step returns	3
Mini-batch size	1024
Discount (γ)	0.99
Optimizer	Adam
Learning rate	10^{-4}
Agent update frequency	2
Critic target EMA rate (τ_Q)	0.01
Features dim.	50
Hidden dim.	1024
Exploration stddev clip	0.3
Exploration stddev value	0.2
Number of frames per episode	1×10^3
Number of online exploration frames	up to 1×10^6
Number of offline planning frames	1×10^5
Critic network	$(\lvert O\rvert + \lvert A\rvert) \to 1024 \to$ LayerNorm $\to$ Tanh $\to 1024 \to$ RELU $\to 1$
Actor network	$\lvert O\rvert \to 50 \to$ LayerNorm $\to$ Tanh $\to 1024 \to$ RELU $\to$ action dim

F.2 Details of offline training algorithm

After collecting the dataset $\mathcal{D}$, we call a reward oracle to label the reward r for any triplet $(s, a, s') \in \mathcal{D}$. Then the DDPG algorithm is called to learn the optimal policy. For the fair comparison with other benchmark algorithm, we do not add weighted regression in the planning phase, thus the algorithm stays the same with the one presented in URLB, as stated in Algorithm 3

F.3 Hyper-parameters

We present a common set of hyper-parameters used in our experiments in Table 3. And we list individual hyper-parameters for each method in table 4. All common hyper-parameters and individual hyper-parameters for baseline algorithms are the same as what is used in Laskin et al. (2021) and its implementations.

F.4 Ablation Study

F.4.1 Learning Processes

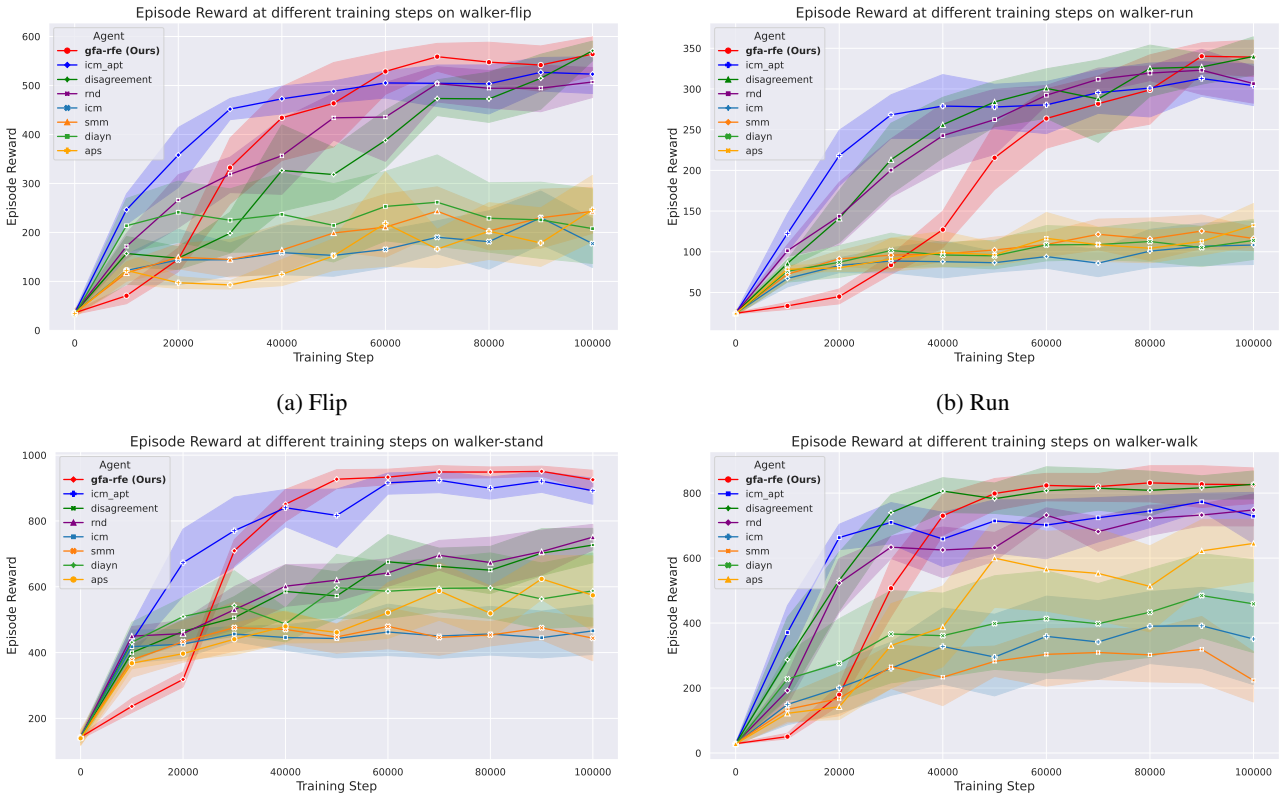
Figures 2 and 3 illustrate the episode rewards for each algorithm across training steps for various tasks, demonstrating that the performance of our algorithm (Algorithm 1) ranks among the top tier in all tasks.

F.4.2 Numbers of Exploration Episodes

Figures 4 and 5 show the episode rewards for top-performing algorithms, including our algorithm (`GFA-RFE`), RND, Disagreement, and APT, across varying numbers of exploration episodes for different tasks. Notably, `GFA-RFE` competes with these leading unsupervised algorithms effectively, matching their performance across a range of exploration episodes.

Table 4. Hyper-parameters of each algorithm.

GFA-RFE	Value
Ensemble size	10
Exploration bonus	2
Exploration ϵ	0.2
ICM hyper-parameter	**Value**
Reward transformation	$\log(r + 1.0)$
Forward net arch.	$(\lvert O\rvert + \lvert A\rvert) \to 1024 \to 1024 \to \lvert O\rvert$ ReLU MLP
Inverse net arch.	$(2 \times \lvert O\rvert) \to 1024 \to \lvert A\rvert$ ReLU MLP
Disagreement hyper-parameter	**Value**
Ensemble size	5
Forward net arch:	$(\lvert O\rvert + \lvert A\rvert) \to 1024 \to 1024 \to \lvert O\rvert$ ReLU MLP
RND hyper-parameter	**Value**
Representation dim.	512
Predictor & target net arch.	$\lvert O\rvert \to 1024 \to 1024 \to 512$ ReLU MLP
Normalized observation clipping	5
APT hyper-parameter	**Value**
Representation dim.	512
Reward transformation	$\log(r + 1.0)$
Forward net arch.	$(512 + \lvert A\rvert) \to 1024 \to 512$ ReLU MLP
Inverse net arch.	$(2 \times 512) \to 1024 \to \lvert A\rvert$ ReLU MLP
k in NN	12
Avg top k in NN	True
SMM hyper-parameter	**Value**
Skill dim.	4
Skill discrim lr	10^{-3}
VAE lr	10^{-2}
DIAYN hyper-parameter	**Value**
Skill dim	16
Skill sampling frequency (steps)	50
Discriminator net arch.	$512 \to 1024 \to 1024 \to 16$ ReLU MLP
APS hyper-parameter	**Value**
Reward transformation	$\log(r + 1.0)$
Successor feature dim.	10
Successor feature net arch.	$\lvert O\rvert \to 1024 \to 10$ ReLU MLP
k in NN	12
Avg top k in NN	True
Least square batch size	4096



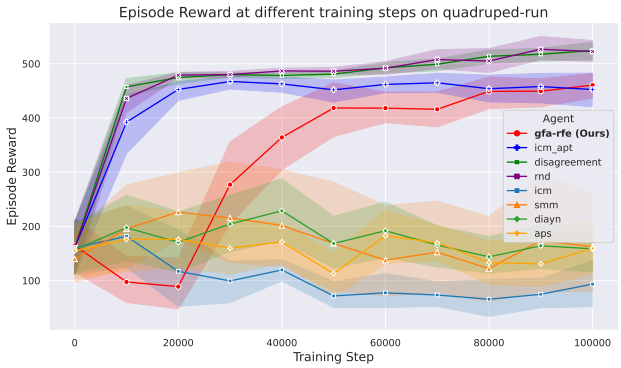
(a) Flip

(b) Run

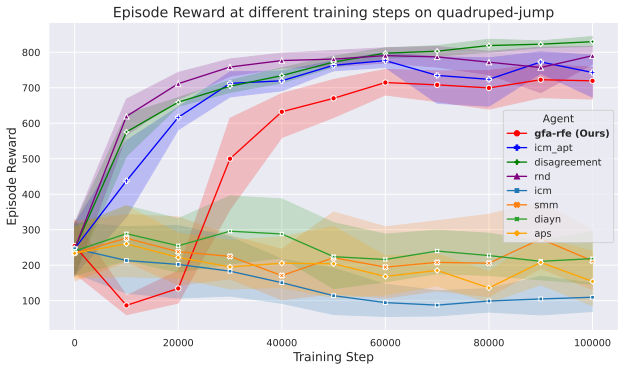
(c) Stand

(d) Walk

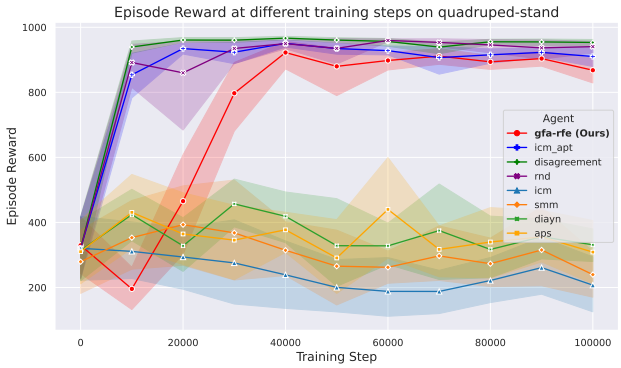
Figure 2. Episode reward at different offline training steps for different tasks for the *walker* environment: (2a): *walker-flip*; (2b): *walker-run*; (2c) *walker-stand*; (2d) *walker-walk*.



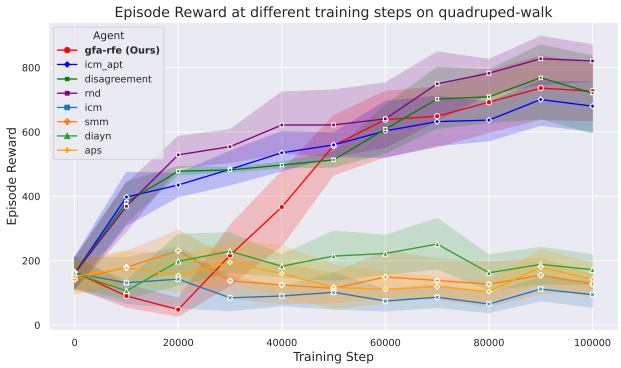
(a) Run



(b) Jump

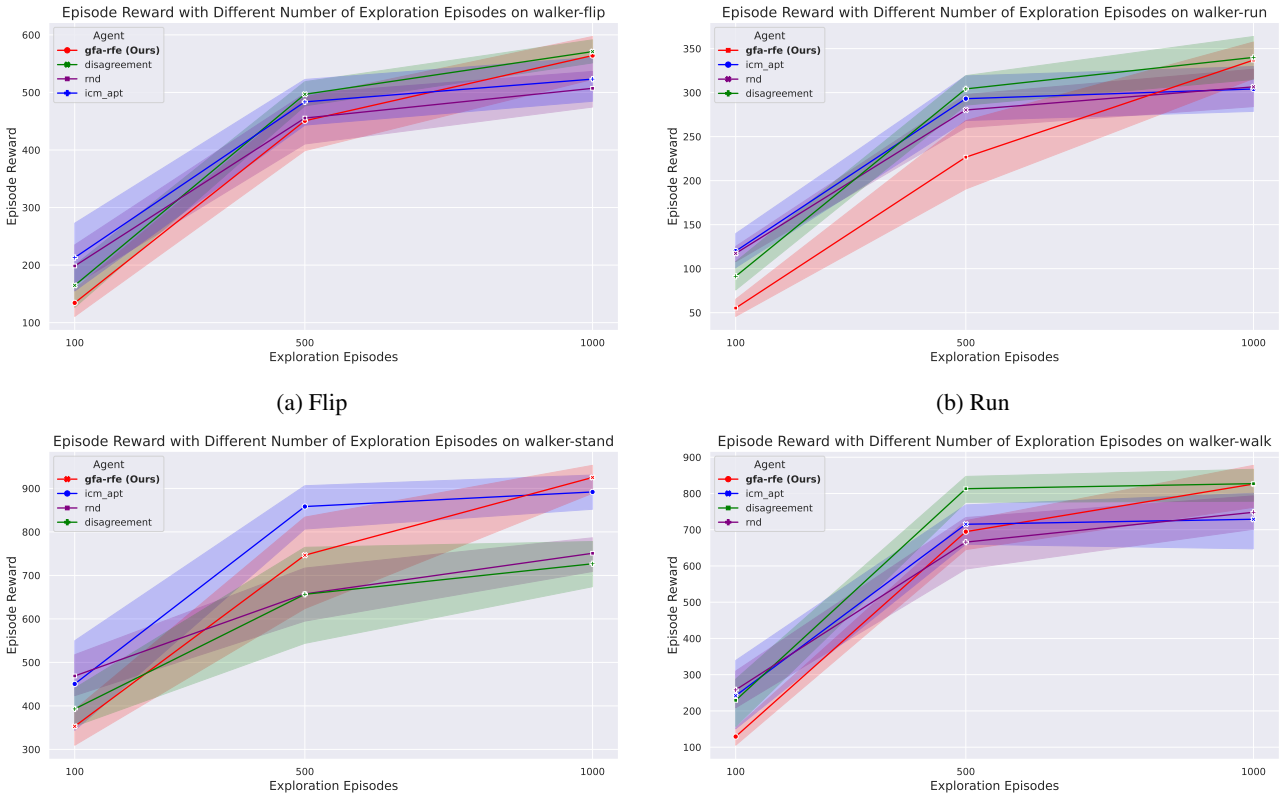


(c) Stand



(d) Walk

Figure 3. Episode reward at different offline training steps for different tasks for the *quadruped* environment: (3a): *quadruped-flip*; (3b): *quadruped-run*; (3c) *quadruped-stand*; (3d) *quadruped-walk*.



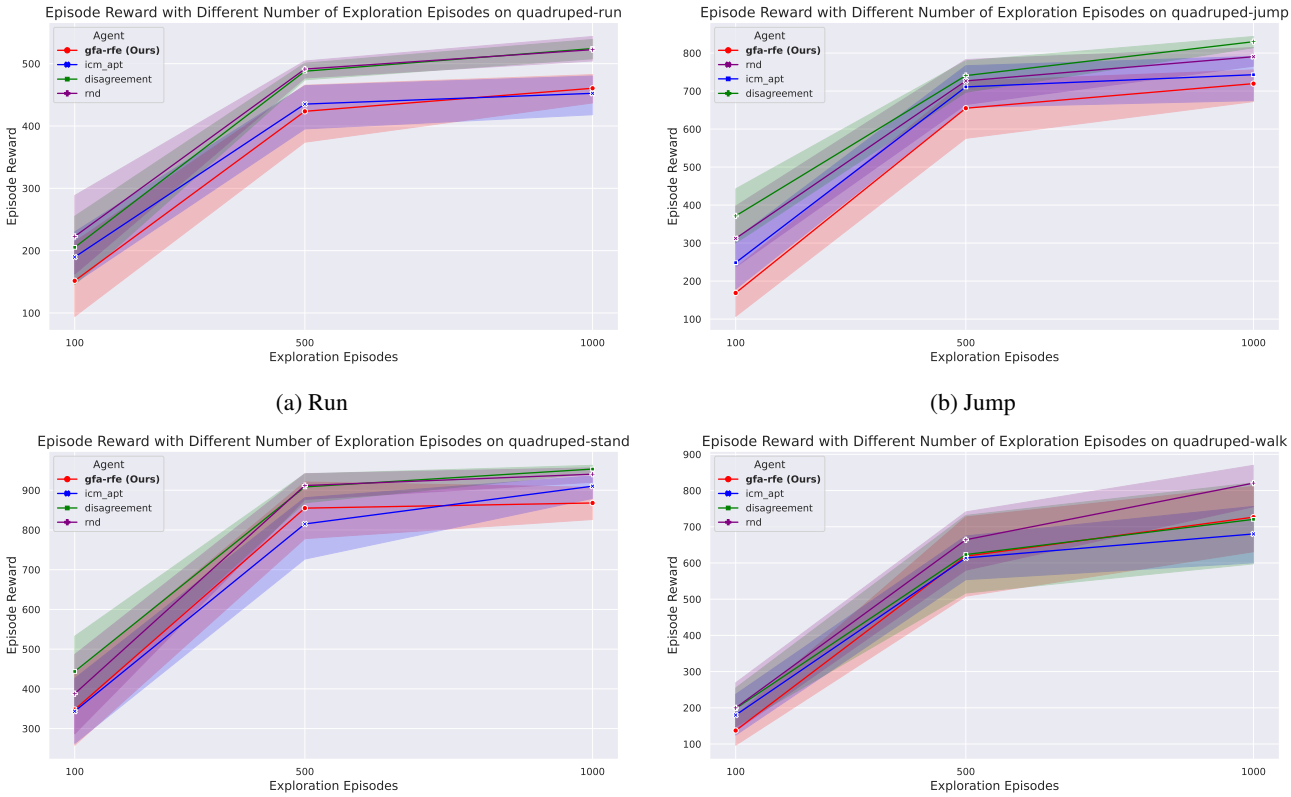
(a) Flip

(b) Run

(c) Stand

(d) Walk

Figure 4. Episode reward with different numbers of exploration episodes for different tasks for the *walker* environment: (4a): *walker-flip*; (4b): *walker-run*; (4c) *walker-stand*; (4d) *walker-walk*.



(a) Run

(b) Jump

(c) Stand

(d) Walk

Figure 5. Episode reward with different numbers of exploration episodes for different tasks for the *quadruped* environment: (5a): *quadruped-flip*; (5b): *quadruped-run*; (5c) *quadruped-stand*; (5d) *quadruped-walk*.