The Horizon Threshold in Cooperative Multi-Agent Reward-Free Exploration

Idan Barnea
Tel Aviv University
idan1id@gmail.com

Orin Levy
Tel Aviv University
orinlevy@mail.tau.ac.il

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

Abstract

We study cooperative multi-agent reinforcement learning in the setting of reward-free exploration, where multiple agents jointly explore an unknown MDP in order to learn its dynamics (without observing rewards). We focus on a tabular finite-horizon MDP and adopt a phased learning framework. In each learning phase, multiple agents independently interact with the environment. More specifically, in each learning phase, each agent is assigned a policy, executes it, and observes the resulting trajectory. Our primary goal is to characterize the tradeoff between the number of learning phases and the number of agents, especially when the number of learning phases is small.

Our results identify a regime change governed by the horizon H. When the number of learning phases equals H, we present a computationally efficient algorithm that uses only $\tilde{O}(S^6H^6A/\epsilon^2)$ agents to learn the dynamics up to value equivalence, i.e., to obtain a model that yields an ϵ-optimal policy for any reward function. We complement our algorithm with a lower bound showing that any algorithm restricted to $\rho < H$ phases requires at least $A^{H/\rho}$ agents to achieve constant accuracy. Thus, we show that having $\Theta(H)$ learning phases is both necessary and sufficient when restricting the number of agents to be polynomial.

1 Introduction

Modern learning systems increasingly rely on multiple agents that cooperate to explore and learn about an unknown dynamic environment. Such cooperative multi-agent settings arise naturally in robotics and distributed control [26], as well as in large language model (LLM) fine-tuning, where a central challenge lies in constructing a high-quality dataset. This dataset is typically collected once and subsequently reused to train models across a wide range of downstream objectives (see, e.g., [6, 19]).

The above scenarios motivate the study of *cooperative multi-agent reinforcement learning (RL)* [20] (sometimes called concurrent RL [7]), in which multiple agents interact with an unknown environment in parallel. This literature extends the classical RL framework based on tabular finite-horizon Markov decision processes (MDPs) to settings where several agents operate on independent copies of the same MDP and share the information they collect.

Finite-horizon MDPs are defined by finite state and action spaces, initial state, transition dynamics and reward function. The action selection strategy of an agent is referred to as *policy*. Each single-agent episode is defined by H interaction steps with the MDP, described by a single *trajectory*. At each step, every agent observes the current state it reaches and selects an action according to its own policy, which may differ across agents. The agent then observes a reward associated with the current state and action and transitions to a next state. All agents experience rewards and transitions governed by the same reward function and transition dynamics of the shared MDP. The learning algorithm

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

has access to the trajectories generated by all agents and can leverage this aggregated experience to update the agents' policies.

However, in many realistic settings as described earlier, the reward function is not specified during data collection, or multiple reward functions may later be of interest for the same environment. This naturally motivates adopting the *reward-free exploration* setup [16].

In reward-free exploration, the goal is to learn the transition dynamics of the environment itself. Rather than optimizing the environment's particular cumulative reward collected during an episode, the learner ignores rewards and focuses on collecting data that enables an accurate estimation of the underlying dynamics.

Crucially, the learned model should be accurate enough so that, for any reward function revealed after data collection, it can be used to reliably evaluate and optimize policies. In particular, planning with the learned dynamics should yield an accurate approximation of the optimal expected cumulative reward, i.e., the *value* of the best policy under the true dynamics of the environment and the given reward function. Since interaction with the environment is often costly, a central objective in reward-free exploration is to achieve this guarantee using as few trajectories as possible.

So far we have viewed the many agents as parallel workers controlled by a single owner. One can also view the problem from a different angle, in which the agents are distinct parties, each with its own interests. Cooperation in reward-free exploration need not stem from a shared, known objective: in many realistic settings the agents do not yet know exactly what they want. Each may have a different, partially specified, or still-evolving task in mind, and the relevant rewards will only be revealed after data collection. Rather than each agent exploring on its own for an objective it cannot yet state, the agents have a common interest in jointly exploring the environment.

In this work, we study the novel setting of *cooperative multi-agent reward-free exploration*; In which, multiple agents explore the environment in parallel under the coordination of a centralized reward-free exploration algorithm, with the shared objective of efficiently learning the underlying dynamics. To model parallel interaction, we gather parallel episodes into *learning phases*, where a learning phase consists of a round of simultaneous m single-agent episodes in which each of the m agents interacts with an independent environment and collects a trajectory without observing the rewards. The centralized learning algorithm aggregates the data collected by all agents in each phase to update its estimate of the environment dynamics and the agents' exploration policies in the next phase.

In this setting, learning efficiency is governed by two key resources. The first is the *parallel time*, measured by the number of learning phases, which determines the distributed computation time required for data collection. The second is the *agent complexity*, measured by the number of agents interacting with the environment in each learning phase, which captures the degree of parallelism available to the learning algorithm. These two resources are inherently coupled, giving rise to a fundamental trade-off between parallel time and agent complexity. At one extreme, when parallel time is unlimited, reward-free exploration can be done sequentially using a single agent (recovering the classical reward-free literature, e.g., [16, 23, 18]). At the other extreme, aggressively limiting parallel time, e.g., to a constant number of learning phases, requires deploying an exponential number of agents (as shown by our lower bound, presented in the sequel).

Our contributions. We study cooperative multi-agent reward-free exploration in tabular Markov decision processes (MDPs) with S states and A actions, where each episode has a finite horizon of length H. Our central focus is understanding the trade-off between *parallel time*, measured by the number of learning phases ρ, and *agent complexity*, measured by the number of agents m in each learning phase.

Our results reveal a *regime change in this trade-off at the horizon scale* H. Specifically, when $\rho = H$, reward-free exploration can be achieved efficiently using a polynomial number of agents. In contrast, when the number of learning phases is reduced below the horizon, i.e., $\rho < H$, any reward-free algorithm necessarily requires the number of agents to grow exponentially in H/ρ. Concretely, our main technical contributions are:

- **Upper bound.** We present `H-MARFE`, a computationally efficient cooperative multi-agent reward-free exploration algorithm that learns the unknown dynamics, up to value equivalence, in exactly $\rho = H$ phases. We prove that, with $m = \tilde{O}(S^6 H^6 A/\epsilon^2)$ agents deployed per phase, the learned model yields an ϵ-optimal policy for any given reward function.

- **Lower bound.** We prove that reducing the number of learning phases significantly below the horizon fundamentally limits parallel efficiency. In particular, any algorithm that uses $\rho < H$ learning phases must deploy at least $m = \Omega\left(A^{H/\rho}/\rho\right)$ agents. This lower bound shows that an order of H learning phases is essential to avoid exponential agent complexity.

Our analysis introduces a novel technique: layer-wise exploration that scales estimation accuracy with reachability, departing from standard regret minimization approaches. Unlike classical regret minimization algorithms (or bonus-based algorithms), which are inherently sequential, our approach is parallelizable. It guarantees low additive error in the occupancy measure difference, which simultaneously ensures accurate value estimation for any reward function and enables the computation of effective exploration directly from the estimated model.

1.1 Related Work

Reward-free. The reward-free model [9] was formally formulated by Jin et al. (2020), who established a sub-optimal sample complexity bound. This bound was subsequently tightened [18, 23]. These works are inherently sequential[1], and it is unclear how their proposed techniques can be parallelized. Qian et al. (2024) study reward-free exploration for contextual MDPs under efficient oracle access to a finite class of dynamics. Applying their algorithm to our setting would require access to an infinite class containing all possible dynamics, which is clearly unlearnable. Reward-free exploration for linear MDPs has been studied [32, 34, 31, 14], but these works are sequential in nature and cannot be easily parallelized. Additional work addressing reward-free exploration include Block-MDP and low-rank settings [1, 35, 5, 24, 2], and other assumptions [37, 4, 25].

Cooperative Multi-Agent RL. In the cooperative multi-agent setting [20]—also known as concurrent RL [7, 8]—multiple agents interact with the environment simultaneously. This setting is also referred to as multi-batch or deployment-efficient RL, and is closely related to the low-switching setting, as the number of policy switches corresponds to the number of parallel phases. Cooperative multi-agent was studied for regret minimization in the tabular setting [20], for linear MDPs [13], and for linear-quadratic dynamical systems [3]. Since regret minimization algorithms (and also bonus-based algorithms) are designed for sequential decision-making, their learning-phase complexity scales linearly with the number of episodes.

Prior empirical works on cooperative multi-agent RL (e.g., [42, 12, 7, 8, 11]) focus on empirical evaluation of reward-based algorithms and not on theoretical analysis. Beyond the distinction between theory and empirical evaluation, direct comparison is further complicated by the fact that our work targets reward-free exploration. For instance, the A3C algorithm [42] relies on sufficient inherent randomization in the rewards and dynamics of the MDP. This allows them to significantly simplify the exploration perspective. Note that this assumption breaks down in deterministic MDPs. In our lower-bound construction with a sparse reward, A3C requires exponentially many (2^H) learning phases before observing any reward, whereas our algorithm learns the dynamics in only H phases with a polynomial number of agents. A similar issue arises in *seed sampling* [7] and DPPO [12].

Qiao et al. (2022) guarantee reward-free learning with $O(SHA)$ switches. In contrast, our algorithm requires only H learning phases (where switches and phases are interchangeable notions). Prior work has also addressed regret minimization in the cooperative multi-agent setting with a focus on high concurrency [36, 39, 41], as well as linear MDPs with deployment costs [15, 40], where the latter achieving deployment cost H and sample complexity $O(d^9H^{15}/\epsilon^5)$.

Additional related work characterizes horizon effects, mostly in the context of off-policy evaluation [30, 21, 38, 17, 10].

2 Preliminaries

We study *reward-free exploration in a cooperative multi-agent reinforcement learning setting*. We begin by defining the single-agent MDP and the notion of reward-free exploration, which will later be extended to the multi-agent case.

[1] Jin et al. (2020) rely on regret-based algorithms to construct exploration policies, and regret guarantees fundamentally apply only in sequential settings; [18, 23] use bonuses to guarantee exploration.

Markov Decision Process. An episodic finite-horizon *Markov Decision Process* (MDP) $\mathcal{M}$ is formally defined as a tuple $\mathcal{M} = (\mathcal{S}, s_0, \mathcal{A}, H, P, r)$ where $\mathcal{S}$ is a finite state space with S states, $s_0 \in \mathcal{S}$ is the initial state[2], $\mathcal{A}$ is a finite action space contains A actions, H is the episode horizon length, $P = \{P_h\}_{h=0}^{H-1}$ is the transition dynamics, and $r = \{r_h\}_{h=0}^{H-1}$ is the reward function.

The transition dynamics P are stochastic and unknown. They specify the transition probabilities at each timestep $h \in [H-1]$ (where $[H-1] := \{0, \ldots, H-1\}$). At each timestep h, for a given state s and action a, the next state s' is drawn according to $P_h(s' \mid s, a)$.

The expected reward at timestep h is given by $r_h(s,a) = \mathbb{E}\left[R_h(s,a)\right]$, where $R_h(s,a) \in [0,1]$ is a random variable sampled from an unknown reward distribution associated with (h,s,a).

An episode is a single run of interaction with the MDP. The sequence of states visited, actions chosen and reward observed defines the trajectory observed in the episode, $(s_0, a_0, r_0, s_1, a_1, r_1, \ldots, s_H)$, where s_h, a_h, r_h denote state, action and reward observed at timestep h.

Policies and values. A (deterministic Markovian) *policy* defines a mapping from timestep h and state s to an action. i.e., $\pi : \mathcal{S} \times [H-1] \to \mathcal{A}$. We denote by $\pi_h(s) = \pi(s,h)$ the action selected by π for state s at timestep h. For MDP $\mathcal{M}$ with dynamics P and reward r, the value of a policy π from the initial state s_0 is given by $V_{P,r}^{\pi} := \mathbb{E}_{P,\pi}\left[\sum_{h=0}^{H-1} r_h(s_h, a_h) \mid s_0\right]$. We explicitly index the value function by (P,r) to emphasize that our focus is on learning the dynamics. The *optimal policy* $\pi^\star$ satisfies $\pi^\star \in \arg\max_\pi V_{P,r}^{\pi}$. We note that there always exists an optimal deterministic policy [27]. The optimal value for P, r is denoted by $V_{P,r}^{\star} := V_{P,r}^{\pi^\star}$. A policy π is ϵ-*optimal* if it holds that $V_{P,r}^{\star} - V_{P,r}^{\pi} \le \epsilon$. In addition, for any policy π, the implied transition matrix, denoted $P_h^\pi \in \mathbb{R}^{S \times S}$, is a row stochastic matrix where $P_h^\pi[s, s'] = P_h(s' \mid s, \pi(s))$.

Occupancy measures. For a policy π and dynamics P, we define the occupancy measure at timestep h as the joint distribution over states and actions induced by π under P. Formally, $q_h(s, a | \pi, P) := \mathbb{P}\left(s_h = s, a_h = a | P, \pi\right)$. We also denote $q_h(s|\pi, P) := \mathbb{P}\left(s_h = s | P, \pi\right)$, which is the probability to visit a state at timestep h. Clearly, $q_h(s, a|\pi, P) = q_h(s|\pi, P)\mathbb{P}\left([\pi_h(s) = a]\right)$. Occupancy measures capture the visitation frequencies of state-action pairs along an episode and thus define a distribution over them.

Reward-free exploration is the task of learning an accurate estimate of the dynamics P, which can later be used to compute an ϵ-optimal policy for any given reward function. In the reward-free exploration phase, the rewards are not observed by the learners, and the focus is to learn an accurate approximation of the dynamics. Formally, let $\widehat{P}$ denote the estimated dynamics learned by a reward-free algorithm when interacting with an unknown environment $\mathcal{M}$. For any reward function r, let $\widehat{\pi}_r := \arg\max_\pi V_{\widehat{P},r}^{\pi}$. The goal of the reward-free learning algorithm is to produce an estimate $\widehat{P}$ such that, with high probability, $V_{P,r}^{\star} - V_{P,r}^{\widehat{\pi}_r} \le \epsilon$.

Cooperative multi-agent RL. We consider a cooperative multi-agent reinforcement learning (MARL, see [33]) setting in which multiple agents interact with a shared environment and share information. We assume *fresh randomness* across agents, i.e., during each learning phase, all random variables are i.i.d sampled for each agent. In particular, even when multiple agents take identical actions in identical states at the same timestep, they observe independently sampled next state.

A learning algorithm in this setting operates over ρ learning phases, each phase decomposed into m episodes of interactions, one for each agent. Formally, in each phase $i \in [\rho]$, the algorithm deploys all m agents to interact with the environment in parallel. Each agent executes a single episode according to its assigned policy, and there is no communication or interaction between agents during the phase. At the end of each phase, the algorithm aggregates the trajectories collected by all agents and updates the policies used in subsequent phases.

Note we distinguish the agent complexity from sample complexity. Sample complexity is the total number of environment interactions required by the algorithm, while agent complexity is the required number of agents in each learning phase. They are closely related: multiplying the agent complexity by the number of learning phases yields the sample complexity, but the reverse is subtler, since a sequential algorithm may not exploit parallelism across agents.

[2]Without loss of generality, we assume an initial-state distribution at the first timestep.

Protocol 1 Interaction Protocol: Cooperative Multi-Agent for Reward-Free Exploration

for each learning phase: $i = 0, \ldots, \rho - 1$ **do**
  for each agent $j \in [m]$ in parallel **do**
    Run policy $\pi_{j,i}$ and observe the trajectory.
  end for
end for
Construct estimated dynamics $\widehat{P}$, and output it.

Cooperative Multi-agent RL for Reward-Free Exploration. We focus on the *reward-free exploration* setting, where agents do not observe the rewards of the environment and their goal is to explore the dynamics in parallel to construct an accurate estimate of it. As in the single-agent reward-free setting, policies are designed to maximize information acquisition rather than cumulative reward. The overall framework for cooperative multi-agent reward-free exploration is presented in Protocol 1.

3 Upper Bound

Our goal is to accelerate reward-free exploration by leveraging multiple agents to reduce the number of learning phases, while keeping the total number of agents polynomial in S, A, H, and ϵ^{-2}. As discussed before, this naturally induces the parallel time versus agent complexity tradeoff. As shown by our lower bound (see Section 4), at least an order H learning phases is necessary for efficient learning. Accordingly, we design an algorithm that operates in exactly H learning phases and requires only a polynomial number of agents.

We note that agent complexity at the scale of $O(1/\epsilon^3)$ can be achieved using a simple construction. We sketch the idea here. In phase i, we explore the timestep-i dynamics P_i. We explore each state–action pair (s, a) by assigning to S/ϵ^3 agents a policy that maximizes the probability of reaching this state–action pair (s, a). If at least $1/\epsilon^2$ agents reach (s, a), we obtain sufficiently many samples to accurately estimate their transitions. Otherwise, it means the probability to visit them is lower than ϵ/S, thus they contribute at most ϵ to the overall estimation error. This yields an ϵ-accurate estimation of the dynamics using $O(1/\epsilon^3)$ agents. The main technical challenge is to achieve the same guarantee using only $O(1/\epsilon^2)$ agents.

Achieving optimal ϵ-dependency within $O(H)$ phases presents several technical challenges. Our algorithm exploits the fact that states differ in their reachability - the maximum probability of reaching a state over all possible policies. A key observation is that exploration effort can be scaled with reachability, allowing higher errors when estimating the next-state probabilities. To leverage this insight, as detailed in Section 3.1, we construct a model with an additive error relative to the true model, i.e., $\|q_h(\cdot \mid \pi, \widehat{P}) - q_h(\cdot \mid \pi, P)\|_1 \leq \alpha h$, where $\widehat{P}$ is the estimation and α scales with ϵ.

This structure gives rise to a second, more subtle challenge. Even when agents are dispatched with policies specifically designed to explore low-reachability states, only a few may actually reach them. For example, when a few agents arrive at states with reachability of α/S, and when there are $\sim S$ such states at timestep h, the estimation error at timestep $h + 1$ can accumulate an additional factor of α. This causes the error to double at each step, leading to exponential growth in the total error.

A third challenge is common to all reward-free algorithms. In each phase, the true dynamics are unknown, yet we must compute policies that explore well. Allowing estimation accuracy to scale with reachability does not affect the value estimates, but it directly affects how the next exploring policies are computed.

Since we only have access to estimated dynamics, we must compute policies that explore effectively despite this uncertainty. To handle both the estimation error and the drift simultaneously, we design the algorithm to target an intermediate, auxiliary dynamics as its goal. Algorithm decisions are evaluated against this auxiliary dynamics, which serves as a reliable proxy for the true dynamics while remaining robust to drift from low-reachability states.

Since the occupancy measures under this auxiliary dynamics differ from the true dynamics by only an additive error at each step h, policies that explore effectively with respect to it will also explore effectively in the true dynamics. This additive structure allows us to leverage a key observation

Algorithm 2 H-PHASE MULTI-AGENT REWARD-FREE EXPLORATION (H-MARFE)

Input: m agents; $\rho = H$ phases; $\beta > 0$.
for phase $i = 0, \ldots, H-1$ **do**
  $\widehat{\pi}^{i,s} \in \arg\max_\pi q_i(s|\pi, \{\widehat{P}_k\}_{k=0}^{i-1})$.
  $\widehat{\mathcal{S}}_i^\beta = \{s \in \mathcal{S} \mid q_i(s \mid \widehat{\pi}^{i,s}, \{\widehat{P}_k\}_{k=0}^{i-1}) \geq \beta\}$.
  for each agent $j \in [m]$ **do**
    Select state and action with uniform distribution $(s,a) \in \widehat{\mathcal{S}}_i^\beta \times \mathcal{A}$.
    Run policy $\widehat{\pi}^{i,s}$ and choose action a on timestep i and state s.
    Observe the trajectory Obs_j.
  end for
  $\widehat{P}_i$: empirical dynamics for $\widehat{\mathcal{S}}_i^\beta$ using $\{Obs_j\}_{j\in[m]}$;
For $s \notin \widehat{\mathcal{S}}_i^\beta$, transition deterministically to s_{sink}.
end for
Output $\widehat{P} = \{\widehat{P}_i\}_{i=0}^{H-1}$.

underlying H-MARFE: it suffices to control the *policy-weighted error* for any policy π — namely, $\|q_h(\cdot \mid \pi, \hat{P})(P_h^\pi - \hat{P}_h^\pi)\|_1$ — to obtain ϵ-accurate estimation of the value.

More concretely, Algorithm H-MARFE (algorithm 2) gets as input a reachability parameter $\beta > 0$ which defines in each learning phase the group of reachable states. Then, the algorithm operates in H multi-agent learning phases. In each phase $i \in [H-1]$, the algorithm explores the dynamics at timestep i (to approximate the transition probability $P_i(\cdot|s_i, a_i)$ to s_{i+1}). Meaning, at phase i, the algorithm updates the approximation only for transitions at timestep i, yields an approximation denoted $\widehat{P}_i$ using the previously-learned dynamics estimation $\{\widehat{P}_j\}_{j=0}^{i-1}$ (which, when clear from the context we denote as $\widehat{P}$).

To collect samples to approximate P_i, the algorithm assigns each agent a random state-action pair to explore. For each (s,a), the algorithm computes a policy $\hat{\pi}^{i,s,a}$ that maximizes the probability of reaching state s at timestep i and taking action a, under the current estimated dynamics $\{\widehat{P}_k\}_{k=0}^{i-1}$.

That is, let $\widehat{\pi}^{i,s} \in \arg\max_\pi q_i(s \mid \pi, \{\widehat{P}_k\}_{k=0}^{i-1})$ and let $\widehat{\pi}^{i,s,a}(s')$ be $\widehat{\pi}^{i,s}(s')$ for $s' \neq s$ and $\widehat{\pi}^{i,s,a}(s') = a$ for $s' = s$. This policy can be computed efficiently using standard planning algorithms. The agents that were assigned (s,a) then execute $\widehat{\pi}^{i,s,a}$ and collect trajectories, which are used to construct empirical estimates of the transition probabilities from timestep i to timestep $i+1$, as described next.

To guarantee the quality of the learned dynamics, we restrict exploration to states that are sufficiently reachable. More formally, we define the set of *active states* at timestep i as the states where their maximum reachability probability under $\widehat{P}$ is at least β: $\widehat{\mathcal{S}}_i^\beta = \left\{s \in \mathcal{S} \;\middle|\; q_i(s \mid \widehat{\pi}^{i,s}, \widehat{P}) \geq \beta\right\}$.

We now describe how the empirical dynamics $\widehat{P}$ are constructed. We introduce an additional absorbing state, denoted by s_{sink}, which captures transitions to states outside the active set of timestep i.

Formally, for transitions at timestep i, we define $\widehat{P}_i(s' \mid s,a) = N_i(s,a,s')/N_i(s,a)$ if $s \in \widehat{\mathcal{S}}_i^\beta$ and $\mathbb{1}[s' = s_{\text{sink}}]$ otherwise, where $N_i(s,a,s')$ denotes the number of agents that in phase i reach state s at timestep i, take action a, and transition to state s'. The quantity $N_i(s,a) = \sum_{s'} N_i(s,a,s')$ is the total number of agents that reach state s at timestep i and choose action a. In each phase i, we use the previously computed dynamics $\{\widehat{P}\}_{k=0}^{i-1}$ to compute the set $\widehat{\mathcal{S}}_i^\beta$ and the played policies $\widehat{\pi}^{i,s}$ as described earlier. See Algorithm 2 for more details.

Below we give an informal statement of our main result (see Theorem A.36 for formal statement).

Theorem 3.1. *Let any $\epsilon, \delta \in (0,1)$. Run H-MARFE (Algorithm 2) with $\beta = \epsilon/(2H^2S)$ and*

$$m = O\left(\frac{S^5H^6A(\log(SHA/\delta) + S)}{\epsilon^2}\right)$$

agents. Then, with probability at least $1-\delta$, for any reward function r it holds that $V_{\widehat{P},r}^\star - V_{\widehat{P},r}^{\widehat{\pi}_r} \leq \epsilon$, where $\widehat{\pi}_r \in \arg\max_\pi V_{\widehat{P},r}^\pi$.

Discussion. Our algorithm is computationally efficient and operates in exactly H learning phases and therefore achieves an optimal (see Section 4 for the lower bound) parallel time, by design. As Theorem 3.1 states, we obtain the result with a polynomial number of agents, as desired. However, while the ϵ dependency is optimal [16], the polynomial dependence on S and H is higher than that of state-of-the-art single-agent reward-free algorithms, such as [23], which achieve a sample complexity of S^2H^3A/ϵ^2. We note that it remains an open question whether such higher dependencies are an inherent cost of parallelization, or merely an artifact of our analysis. As the focus of our paper is the change of the learning regimes, we leave tightening these dependencies for future work.

Our results also rely on several structural assumptions: finite-horizon tabular MDPs, centralized coordination, independent environment copies across agents, and no within-phase communication. Relaxing these assumptions is an interesting future work direction.

Value equivalence. We emphasize that our guarantee is value-based: we estimate the dynamics only up to value equivalence, rather than uniformly approximating the transition kernel over all state-action pairs. In particular, H-MARFE redirects states with low reachability to the sink state s_{sink}, so their transitions are not estimated at all. Since such states are rarely visited by any policy, this does not affect the accuracy of value estimates, and the learned model $\widehat{P}$ suffices for computing an ϵ-optimal policy for any reward function revealed after exploration.

Computational complexity. Each phase of H-MARFE requires computing, for every target state, the policy $\widehat{\pi}^{i,s}$ that maximizes the probability of reaching it, together with its value $q_i(s \mid \widehat{\pi}^{i,s}, \{\widehat{P}_k\}_{k=0}^{i-1})$, which also determines the active set $\widehat{\mathcal{S}}_i^{\beta}$. This reduces to standard planning via dynamic programming.

Fix a target state s_{goal} at timestep h. Define an auxiliary reward function that equals 1 at s_{goal} at timestep h and 0 everywhere else. The optimal value under this reward is exactly the reachability of s_{goal} at timestep h, and the optimal policy is $\widehat{\pi}^{h,s_{\mathrm{goal}}}$. We compute it by backward induction under the estimated dynamics $\widehat{P}$: for states at timestep h we set $V_h(s) = \mathbb{I}(s = s_{\mathrm{goal}})$, and for $j = h, h-1, \ldots, 1$,

$$V_{j-1}(s) \;=\; \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} \widehat{P}_{j-1}(s' \mid s, a)\, V_j(s').$$

Each backward step costs $O(S^2A)$, so a single target state costs $O(S^2Ah)$. Repeating this for all S states of timestep h costs $O(S^3Ah)$ per phase, and $O(S^3AH^2)$ over all H phases.

3.1 Analysis

In this section, we present the main steps of the upper bound analysis. To achieve high parallelism, we avoid reducing to regret minimization techniques, as these are inherently designed for sequential execution.

The proof reduces to bounding the value difference $|V^{\pi}_{P,r} - V^{\pi}_{\widehat{P},r}| \le \epsilon/2$ for every policy π and reward r (see Lemma A.35), which in turn reduces to controlling the gap between occupancy measures induced by $\widehat{P}$ and the true dynamics.

To this end, we introduce an intermediate dynamics P^1, which follows P on the sets $\widehat{\mathcal{S}}_h^{\beta} = \{s \mid q_h(s \mid \widehat{\pi}^{h,s}, \widehat{P}) \ge \beta\}$ of β-reachable states under $\widehat{P}$, and otherwise transits deterministically to s_{sink}.

Key step: policy-weighted error bound. Our goal is to prove by induction on h, that for any policy π

$$\|q_h(\cdot \mid \pi, \{\widehat{P}_j\}_{j=0}^{h-1}) - q_h(\cdot \mid \pi, \{P_j^1\}_{j=0}^{h-1})\|_1 \le \alpha h, \tag{1}$$

where $\alpha = \beta/(3H)$. The core of the proof is the following bound, which controls the policy-weighted transition error between $\widehat{P}$ and P^1. We show the key part of the induction step: assuming the induction hypothesis holds for h (Equation (1)), then for every policy π,

$$\|q_h(\cdot \mid \pi, \{\widehat{P}_j\}_{j=0}^{h-1})(\widehat{P}_h^{\pi} - P_h^{1,\pi})\|_1 \le \alpha. \tag{2}$$

Since P^1 agrees with P on $\widehat{\mathcal{S}}^\beta$, we have $q_h(s \mid \widehat{\pi}^{h,s}, P) \geq q_h(s \mid \widehat{\pi}^{h,s}, P^1)$, so the expected number of agents reaching s at step h is at least $q_h(s \mid \widehat{\pi}^{h,s}, P^1)\, m/(SA)$. An adapted inequality then yields, with high probability, for all h, s, a:

$$\|\widehat{P}_h(\cdot \mid s,a) - P^1_h(\cdot \mid s,a)\|_1 \leq \sqrt{\frac{\log(1/\delta) + S}{\frac{m}{SA}\, q_h(s \mid \widehat{\pi}^{h,s}, P^1)}}. \tag{3}$$

States outside $\widehat{\mathcal{S}}^\beta_h$ contribute nothing to Equation (2), since both dynamics send them to s_{sink}. Fix $s \in \widehat{\mathcal{S}}^\beta_h$. Using $\widehat{P}$ alone, we can compute $\widehat{\pi}^{h,s}$, the policy that maximizes the visitation probability of s under $\widehat{P}$. By the induction hypothesis, $\widehat{\pi}^{h,s}$ also nearly maximizes the visitation probability under P^1: for any policy π, $q_h(s \mid \pi, \widehat{P}) \leq q_h(s \mid \widehat{\pi}^{h,s}, \widehat{P}) \leq q_h(s \mid \widehat{\pi}^{h,s}, P^1) + \alpha h$. From Equation (3) (see Lemma A.19)) it suffices to bound

$$\big(q_h(s \mid \widehat{\pi}^{h,s}, P^1) + \alpha h\big) \cdot \sqrt{\frac{\log(1/\delta) + S}{\frac{m}{SA}\, q_h(s \mid \widehat{\pi}^{h,s}, P^1)}} \leq \frac{\alpha}{S}. \tag{4}$$

The first term reduces to $\sqrt{(q_h(s \mid \widehat{\pi}^{h,s}, P^1)\,(\log(1/\delta) + S)\, SA)/m}$, which is at most $\alpha/(2S)$ by our choice of m. For the second term, since we trimmed states with low reachability, it holds that $q_h(s \mid \widehat{\pi}^{h,s}, P^1) \geq 2\beta/3$. Hence, $1/\sqrt{q_h(s \mid \widehat{\pi}^{h,s}, P^1)} = O(1/\sqrt{\alpha H})$, and m is large enough to yield $\alpha/(2S)$. Combining the two bounds gives (2). The remainder of the induction step is deferred to the appendix (Theorem A.25).

Closing the gap: P^1 approximates P. Note that P^1 is defined with respect to the *reachability under the estimated dynamics* rather than the reachability under the true dynamics. To bridge this gap, we introduce an auxiliary dynamics P^2 and use it to compare P^1 with P. The auxiliary dynamics is defined inductively on the *true* inductive-low-reachability set $\mathcal{S}^2_h := \{s \mid \max_\pi q_h(s \mid \pi, \{P^2_j\}_{j=0}^{h-1}) \geq 2\beta\}$ (where $P^2_0 = P_0$).

Standard arguments show P^2 is close to P. Therefore it remains to show that P^2 is close to P^1. For that it suffices to prove $\mathcal{S}^2_h \subseteq \widehat{\mathcal{S}}^\beta_h$ for all h.

By induction: Assume $s \in \mathcal{S}^2_{h+1}$. Let π be a witness for $s \in \mathcal{S}^2_{h+1}$, so $2\beta \leq q_{h+1}(s \mid \pi, P^2)$. Applying the induction hypothesis, we can show $q_{h+1}(s \mid \pi, P^2) \leq q_{h+1}(s \mid \pi, P^1)$ (Lemma A.27). Here the bound we derived in Equation (1) plays another role. By Equation (1), the occupancy measures under P^1 and $\widehat{P}$ differ by at most $\alpha(h+1) \leq \beta/3$ for any policy, and in particular, this bound holds for the policy π that certifies $s \in \mathcal{S}^2_{h+1}$. Therefore,

$$2\beta \leq q_{h+1}(s \mid \pi, P^2) \leq q_{h+1}(s \mid \pi, P^1) \leq q_{h+1}(s \mid \pi, \widehat{P}) + \beta/3 \leq q_{h+1}(s \mid \widehat{\pi}^{h,s}, \widehat{P}) + \beta/3,$$

where the last inequality is since $\widehat{\pi}^{h,s}$ maximizes the probability of reaching state s at timestep h. Rearranging the terms, we obtain that $s \in \widehat{\mathcal{S}}^\beta_{h+1}$. For more details see Lemma A.28.

Together, these steps establish that $\widehat{P}$ approximates P over all policies and rewards, completing the proof sketch for the upper bound.

Remark 3.2. We believe that our exploration strategy may extend to linear MDPs. Conceptually, one would replace states with feature basis vectors, and the set-visitation measure [31] is analogous to our notion of reachability. However, existing reward-free algorithms for linear MDPs are inherently sequential, as most rely on bonuses or on regret-minimization techniques with synthetic rewards for exploration. Since our framework does not employ standard regret minimization techniques, designing a parallelized counterpart introduces additional challenges. More generally, under general function approximation, one would need a notion of low coverage whose contribution to the estimation error can still be controlled through its limited occupancy, allowing the same cancellation between visitation and confidence that drives our tabular analysis (see Equation (4)).

4 Lower Bound

In this section, we present a lower bound proving that when the number of learning phases is too small (i.e., significantly smaller than H), achieving reward-free guarantees necessarily requires an

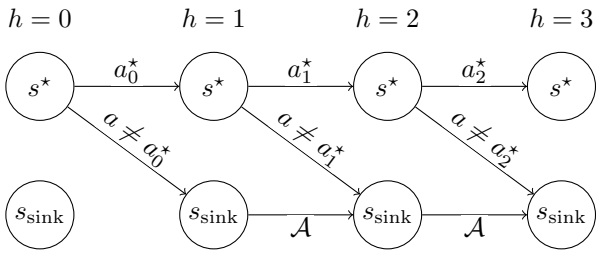


Figure 1: Hidden key for the lower bound: only $a_h^\star$ stays in $s^\star$; otherwise transition to s_{sink}.

exponential number of agents. This result complements our upper bound, which demonstrates that efficient reward-free learning using only a polynomial number of agents is achievable with H learning phases. Taken together, the upper and lower bounds reveal a regime change in learning efficiency at the horizon scale H, separating regimes of polynomial and exponential agent complexity.

Our lower bound is stated next (see section B for full proof).

Theorem 4.1. *Let $\mathtt{A}$ be a cooperative multi-agent algorithm that runs for ρ learning phases with m agents in each phase. Assume it satisfies the reward-free guarantee for $\epsilon = 0.1$ and $\delta = 0.44$, i.e., it outputs dynamics $\widehat{P}$ such that for any reward function r, $\mathbb{P}\left(V_{P,r}^\star - V_{P,r}^{\widehat{\pi}_r} < 0.1\right) > 0.56$, where $\widehat{\pi}_r \in \arg\max_\pi \{V_{\widehat{P},r}^\pi\}$. Then, $\mathtt{A}$ must deploy*

$$m = \Omega\left(\frac{A^{(H-1)/\rho}}{\rho}\right)$$

agents in each learning phase.

To build intuition, we sketch the proof for $\rho = 1$, and defer the proof for general ρ to Theorem B.15.

Proof sketch for $\rho = 1$. Our lower-bound instance is an MDP whose dynamics encode a hidden key of length H, with symbols drawn from the action set $\mathcal{A}$. Intuitively, an algorithm deploying m agents can uncover at most an additional $\log_A(m)$ actions of this key in each learning phase. Hence, to achieve the desired accuracy for a reward function r that assigns a reward of 1 only upon reaching the end of the key and 0 otherwise, the algorithm must therefore uncover the entire key.

More formally, consider the state space $\mathcal{S} = \{s_{\text{sink}}, s^\star\}$, and the initial state is $s_0 = s^\star$. A random key, denoted $a_0^\star, \ldots, a_{H-1}^\star$, is chosen and fixed. The dynamics are defined using this key as follows. For all $h \in [H-1]$, at state $s^\star$, taking action $a_h^\star$ deterministically transitions back to $s^\star$, while taking any other action transitions to s_{sink}. From state s_{sink}, taking any action deterministically transitions back to s_{sink}, meaning that once an agent reaches s_{sink}, it cannot escape. See Figure 1 for a diagram of the dynamics for $H = 3$.

Consequently, when a small number of agents are deployed, recovering the full key, and thus the dynamics, is impossible. To see this, for all $h \in [H]$, we denote by $\mathcal{G}_h \subseteq [m]$ the subset of the agents that reached state $s^\star$ at timestep h. We first show that for all fixed h, $\mathbb{E}\left(|\mathcal{G}_{h+1}|\right) = \mathbb{E}\left(|\mathcal{G}_h|\right)/A$.

Let a_h^i denote the action that agent i played at timestep h. When conditioning on $\mathcal{G}_h$ we obtain

$$\begin{aligned}\mathbb{E}\left(|\mathcal{G}_{h+1}| \mid \mathcal{G}_h\right) &= \sum_{a\in\mathcal{A}} \sum_{i\in\mathcal{G}_h} \mathbb{P}\left(a_h^i = a, a = a_h^\star \mid \mathcal{G}_h\right) \\ &= \frac{1}{A} \sum_{i\in\mathcal{G}_h} \sum_{a\in\mathcal{A}} \mathbb{P}\left(a_h^i = a \mid \mathcal{G}_h, a = a_h^\star\right) = \frac{|\mathcal{G}_h|}{A},\end{aligned}$$

where we use the fact that a_h^i and $a_h^\star$ are independent, and that each agent chooses only one action. Thus $\sum_{a\in\mathcal{A}} \mathbb{P}\left(a_h^i = a \mid \mathcal{G}_h, a = a_h^\star\right) = 1$. Next, we take the expectation over $\mathcal{G}_h$ and combine with the above to obtain

$$\mathbb{E}\left(|\mathcal{G}_{h+1}|\right) = \mathbb{E}\left(\mathbb{E}\left(|\mathcal{G}_{h+1}| \mid \mathcal{G}_h\right)\right) = \frac{\mathbb{E}\left(|\mathcal{G}_h|\right)}{A} = \frac{m}{A^{h+1}},$$

where the last equality can be shown by induction. Thus, when $m \ll A^{H-1}$, there is a constant probability that $\mathcal{G}_H$ is empty; therefore, the algorithm fails to recover the full key. Consequently, for our reward function r (given by $r_h(s,a) = \mathbb{1}(h = H-1,\ s = s^\star,\ a = a^\star_{H-1})$), the optimal policy achieves a value of 1, while the algorithm has a constant probability of failure, yielding a value strictly smaller than 0.9. This contradicts the assumed 0.1-optimality of `A`, proving the theorem. □

Remark 4.2. The horizon dependency we establish reflects a regime change, though the transition is gradual rather than sharp. One can leverage `H-MARFE` to learn in $H/2$ phases. By condensing an MDP with H layers and A actions into one with $H/2$ layers and A^2 actions, the reformulated MDP can be learned in $H/2$ phases (potentially requiring A times more agents). This recovers all even layers within $H/2$ phases. Since every odd layer can be learned in parallel with its subsequent even layer—at the cost of doubling the number of agents—the entire MDP can be learned in $H/2$ phases using $\tilde{O}(S^6H^6A^2/\epsilon^2)$ agents. As our lower bound shows, setting $\rho = H/2$ yields an agent complexity of $\sim A^2/H$. The regime where our lower bound is most significant is where $\rho \ll H$.

Similarly, we can replace 2 with any constant c (assuming for simplicity that c divides H): condensing every c consecutive layers yields an MDP with H/c layers and A^c actions, and the remaining $c-1$ layers in each block are learned in parallel with the subsequent recovered layer, at the cost of a factor c more agents. Hence, $\tilde{O}(S^6H^6A^c/\epsilon^2)$ agents suffice to learn the dynamics, up to value equivalence, in $\rho = H/c$ phases.

On the other hand, setting $\rho = H/c$ in Theorem 4.1 yields a lower bound of $\Omega\big(A^{c(H-1)/H}\, c/H\big)$, whose exponent of A differs from c by only multiplying with $c/H \leq 1$. Thus, the exponential dependence on A^c is unavoidable. In particular, as c increases towards H, i.e., as the number of phases approaches a constant, the number of agents becomes exponential in H. The regime where our lower bound is most significant is therefore $\rho \ll H$.

For $\rho > H$, we do not characterize the precise trade-off. State-of-the-art single-agent algorithms require $\rho \geq \Omega(SH/\epsilon^2)$ phases, and since our guarantees are already polynomial at $\rho = H$, we expect the trade-off in this regime to be even more gradual. Characterizing it precisely is a natural direction for future work.

5 Discussion and Future Work

In this work, we introduce the problem of cooperative multi-agent reward-free exploration, in which a centralized learning algorithm deploys multiple agents in parallel to learn unknown environment dynamics. We identify a fundamental trade-off between parallel time and agent complexity that naturally arises in this setting and analyze it. Our results present a regime change when the number of learning phases scales with H, the horizon of the learned MDP. For H learning phases we show a computationally efficient algorithm that requires $\tilde{O}(S^6H^6A/\epsilon^2)$ agents. In contrast, when the number of learning phases is reduced to $\rho < H$, we prove that any algorithm must deploy at least $\Omega(A^{(H-1)/\rho}/\rho)$ agents. Together, these results demonstrate two distinct learning regimes: one with polynomial and one with exponential agent complexity, determined by the number of learning phases.

Several directions for future work remain. First, reducing the polynomial dependence on S and H in our upper bound — both under general conditions and under additional assumptions — would lead to a tighter characterization of the polynomial regime. Second, extending our results to settings with large or continuous state spaces is another important direction. We hope this work inspires deeper understanding of cooperative multi-agent exploration and further investigation into the fundamental limits of parallel reinforcement learning.

Acknowledgments

This project is supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 882396), by the Israel Science Foundation (1357/24), the Blavatnik foundation at Tel Aviv University and a grant from the Tel Aviv University Center for AI and Data Science (TAD). IB is also supported by the Israeli Council for Higher Education (VATAT) scholarship for outstanding doctoral students in artificial intelligence, and by the Tel Aviv University Center for AI and Data Science (TAD) excellence program for doctoral students in artificial intelligence. OL is also supported by the Google PhD fellowship award (2025).

References

[1] A. Agarwal, S. M. Kakade, A. Krishnamurthy, and W. Sun. FLAMBE: structural complexity and representation learning of low rank mdps. In *Advances in Neural Information Processing Systems*, 2020.

[2] P. Amortila, D. J. Foster, and A. Krishnamurthy. Scalable online exploration via coverability. In *Forty-first International Conference on Machine Learning, ICML*, 2024.

[3] S. M. Asghari, Y. Ouyang, and A. Nayyar. Regret bounds for decentralized learning in cooperative multi-agent dynamical systems. In *Proceedings of the Thirty-Sixth Conference on Uncertainty in Artificial Intelligence, UAI*, 2020.

[4] J. Chen, A. Modi, A. Krishnamurthy, N. Jiang, and A. Agarwal. On the statistical efficiency of reward-free exploration in non-linear RL. In *Advances in Neural Information Processing Systems*, 2022.

[5] Y. Cheng, R. Huang, Y. Liang, and J. Yang. Improved sample complexity for reward-free reinforcement learning under low-rank mdps. In *The Eleventh International Conference on Learning Representations, ICLR*, 2023.

[6] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Y. Zhao, Y. Huang, A. M. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei. Scaling instruction-finetuned language models. *J. Mach. Learn. Res.*, 2024.

[7] M. Dimakopoulou and B. V. Roy. Coordinated exploration in concurrent reinforcement learning. In J. G. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML*, Proceedings of Machine Learning Research. PMLR, 2018.

[8] M. Dimakopoulou, I. Osband, and B. V. Roy. Scalable coordinated exploration in concurrent reinforcement learning. In S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS*, 2018.

[9] B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine. Diversity is all you need: Learning skills without a reward function. In *7th International Conference on Learning Representations, ICLR*, 2019.

[10] D. J. Foster, A. Block, and D. Misra. Is behavior cloning all you need? understanding horizon in imitation learning. In A. Globersons, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. M. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing*, 2024.

[11] J. Hao, T. Yang, H. Tang, C. Bai, J. Liu, Z. Meng, P. Liu, and Z. Wang. Exploration in deep reinforcement learning: From single-agent to multiagent domain. *IEEE Trans. Neural Networks Learn. Syst.*, 2024.

[12] N. Heess, D. TB, S. Sriram, J. Lemmon, J. Merel, G. Wayne, Y. Tassa, T. Erez, Z. Wang, S. M. A. Eslami, M. A. Riedmiller, and D. Silver. Emergence of locomotion behaviours in rich environments. *CoRR*, 2017.

[13] H. Hsu and M. Pajic. Safe cooperative multi-agent reinforcement learning with function approximation. In *7th Annual Learning for Dynamics & Control Conference*, 2025.

[14] P. Hu, Y. Chen, and L. Huang. Towards minimax optimal reward-free reinforcement learning in linear mdps. In *The Eleventh International Conference on Learning Representations, ICLR*, 2023.

[15] J. Huang, J. Chen, L. Zhao, T. Qin, N. Jiang, and T. Liu. Towards deployment-efficient reinforcement learning: Lower bound and optimality. In *The Tenth International Conference on Learning Representations, ICLR*, 2022.

[16] C. Jin, A. Krishnamurthy, M. Simchowitz, and T. Yu. Reward-free exploration for reinforcement learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML*, 2020.

[17] N. Kallus and M. Uehara. Efficiently breaking the curse of horizon in off-policy evaluation with double reinforcement learning. *Oper. Res.*, 2022.

[18] E. Kaufmann, P. Ménard, O. D. Domingues, A. Jonsson, E. Leurent, and M. Valko. Adaptive reward-free exploration. In *Algorithmic Learning Theory*, 2021.

[19] T. Kaufmann, P. Weng, V. Bengs, and E. Hüllermeier. A survey of reinforcement learning from human feedback. *Trans. Mach. Learn. Res.*, 2025.

[20] T. Lancewicki, A. Rosenberg, and Y. Mansour. Cooperative online learning in stochastic and adversarial MDPs. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

[21] Q. Liu, L. Li, Z. Tang, and D. Zhou. Breaking the curse of horizon: Infinite-horizon off-policy estimation. In S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing*, 2018.

[22] S. Mannor, Y. Mansour, and A. Tamar. *Reinforcement Learning: Foundations*. -, 2022.

[23] P. Ménard, O. D. Domingues, A. Jonsson, E. Kaufmann, E. Leurent, and M. Valko. Fast active learning for pure exploration in reinforcement learning. In *Proceedings of the 38th International Conference on Machine Learning, ICML*, 2021.

[24] Z. Mhammedi, A. Block, D. J. Foster, and A. Rakhlin. Efficient model-free exploration in low-rank mdps. In *Advances in Neural Information Processing Systems*, 2023.

[25] S. Miryoosefi and C. Jin. A simple reward-free approach to constrained reinforcement learning. In *International Conference on Machine Learning, ICML*, 2022.

[26] J. Orr and A. Dutta. Multi-agent deep reinforcement learning for multi-robot applications: A survey. *Sensors*, 23, 2023.

[27] M. L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[28] J. Qian, H. Hu, and D. Simchi-Levi. Offline oracle-efficient learning for contextual mdps via layerwise exploration-exploitation tradeoff. In *Advances in Neural Information Processing Systems*, 2024.

[29] D. Qiao, M. Yin, M. Min, and Y. Wang. Sample-efficient reinforcement learning with loglog(t) switching cost. In *International Conference on Machine Learning, ICML*, 2022.

[30] S. Ross and D. Bagnell. Efficient reductions for imitation learning. In Y. W. Teh and D. M. Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, 2010.

[31] A. J. Wagenmaker, Y. Chen, M. Simchowitz, S. S. Du, and K. G. Jamieson. Reward-free RL is no harder than reward-aware RL in linear markov decision processes. In *International Conference on Machine Learning, ICML*, 2022.

[32] R. Wang, S. S. Du, L. F. Yang, and R. Salakhutdinov. On reward-free reinforcement learning with linear function approximation. In *Advances in Neural Information Processing Systems*, 2020.

[33] K. Zhang, Z. Yang, and T. Başar. *Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms*, pages 321–384. Springer International Publishing, 2021.

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

[35] X. Zhang, Y. Song, M. Uehara, M. Wang, A. Agarwal, and W. Sun. Efficient reinforcement learning in block mdps: A model-free representation learning approach. In *International Conference on Machine Learning, ICML*, 2022.

[36] Z. Zhang, Y. Zhou, and X. Ji. Almost optimal model-free reinforcement learning via reference-advantage decomposition. In *Advances in Neural Information Processing Systems*, 2020.

[37] Z. Zhang, S. Du, and X. Ji. Near optimal reward-free reinforcement learning. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[38] Z. Zhang, X. Ji, and S. S. Du. Is reinforcement learning more difficult than bandits? A near-optimal algorithm escaping the curse of horizon. In M. Belkin and S. Kpotufe, editors, *Conference on Learning Theory, COLT 2021, 15-19 August 2021, Boulder, Colorado, USA*, 2021.

[39] Z. Zhang, Y. Jiang, Y. Zhou, and X. Ji. Near-optimal regret bounds for multi-batch reinforcement learning. In *Advances in Neural Information Processing Systems*, 2022.

[40] Z. Zhang, Y. Chen, J. D. Lee, S. S. Du, L. Yang, and R. Wang. Deployment efficient reward-free exploration with linear function approximation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[41] H. Zhao, J. He, and Q. Gu. A nearly optimal and low-switching algorithm for reinforcement learning with general function approximation. In *Advances in Neural Information Processing Systems*, 2024.

[42] Z. Zhao, Y. Ziser, and S. B. Cohen. Layer by layer: Uncovering where multi-task learning happens in instruction-tuned large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP*, 2024.

A Upper bound

In this section we prove our upper bound.

A.1 Definitions

Definition A.1. We denote the value function of policy π under dynamics P and reward r, with $V^{\pi}(s_0; P, r) := V^{\pi}_{P,r}$. The optimal value function is denoted with $V^{\star}(s_0; P, r)$.

Definition A.2. We denote with α the error of the occupancy measures estimation. The theorems will assume that α scales with ϵ to ensure ϵ optimality.

Definition A.3. We define the β parameter to be,

$$\beta := 3\alpha H.$$

This parameter will be used to define the states we ignore in our transition function estimation.

Remark A.4. Without loss of generality, the reward functions considered in the appendix are deterministic. Since our analysis concerns the value function, which depends only on the expected reward, we can restrict attention to deterministic reward functions.

Remark A.5. We generalize the policy definition and allow policies to be stochastic, even though our algorithm `H-MARFE` (Algorithm 2) uses only deterministic policies. Hence, we could restrict the setting to deterministic policies only, but we choose to show that the algorithm returns a good estimate of the dynamics for every stochastic policy. For example, when we compare the dynamics the algorithm estimated with the true dynamics, we need to consider every stochastic policy, and not just deterministic, see for example the proof of Theorem A.25. In other words, we assume $\pi_h(s)$ induces a probability over the actions. The value function is over this randomness as well as the randomness of the MDP.

A.1.1 The estimated dynamics $\widehat{P}$

We define the estimated dynamics, $\widehat{P}$, with the empirical mean. To handle states s where we do not have a good estimation for the transition functions $P_h(\cdot \mid s, \cdot)$, we introduce a virtual sink state s_{sink} and a criterion for reachability. We continue with the empirical probability only when we estimate the state to be reachable with probability at least β, from the dynamics we built in the previous timestep.

Definition A.6. Let $\widehat{q}(s \mid \pi) := q_h(s \mid \pi, \widehat{P})$ be the **occupancy measure** of state $s \neq s_{\text{sink}}$ at timestep h, when policy π played over dynamics $\widehat{P}$. I.e., $q_h(s \mid \pi, \widehat{P})$ is the probability to reach state s at timestep h, under dynamic $\widehat{P}$ with policy π. We also denote $\widehat{q}$ where it is clear what the policy is. Note that we take $\widehat{q}$ over $\mathcal{S}$ and without s_{sink}, just for the ease of notation.

Definition A.7. We denote with $\widehat{\pi}^{h,s}$ the policy that under the estimated dynamics $\widehat{P}$, reaches state s at step h with maximum probability. I.e., $\widehat{\pi}^{h,s} = \arg\max_{\pi}\{q_h(s \mid \pi, \widehat{P})\}$.

Definition A.8. Let $\widehat{\mathcal{S}}^{\beta}_h$ denote the set of states that are empirically β-reachable at timestep h:

$$\widehat{\mathcal{S}}^{\beta}_h := \{s \in \mathcal{S} \mid q_h(s \mid \widehat{\pi}^{h,s}, \widehat{P}) \geq \beta\}. \tag{5}$$

The **transition function** $\widehat{P}_h$ is defined by:

s	s'	$\widehat{P}_h(s' \mid s, a)$
$\widehat{\mathcal{S}}^{\beta}_h$	$\mathcal{S}$	$\frac{N_h(s,a,s')}{N_h(s,a)}$
$\mathcal{S} \setminus \widehat{\mathcal{S}}^{\beta}_h$	s_{sink}	1
s_{sink}	s_{sink}	1
Otherwise		0

Where $N_h(s, a, s')$ is the number of times at phase h an agent was at timestep h, state s, took action a, and reached s'. The total number of visitations is denoted with $N_h(s, a) = \sum_{s' \in \mathcal{S}} N_h(s, a, s')$.

Notation	Meaning
P	True dynamics
$\widehat{P}$	Estimated dynamics; Estimated β-reachable states
P^1	True dynamics; Estimated β-reachable states
P^2	True dynamics; True 2β-reachable states

Table 1: Summary of dynamics notations.

A.1.2 The dynamics P^1

We define an intermediate dynamics, denoted by P^1, which serves as a bridge for comparing the empirical estimates to the true dynamics. This model is constructed timestep-by-timestep by restricting transitions from states that are empirically β-reachable.

Specifically, we continue the dynamics only from states that meet a reachability threshold based on our current estimations. Since this set depends on estimation, it is a random variable.

To account for the probability mass that enters non-reachable states, we introduce a sink state s_{sink}. The state space of P^1 is $\mathcal{S} \cup \{s_{\text{sink}}\}$.

A.1.3 The transition function of P^1

The **transition function** P_h^1 is defined by:

s	s'	$P_h^1(s' \mid s, a)$
$\widehat{\mathcal{S}}_h^\beta$	$\mathcal{S}$	$P_h(s' \mid s, a)$
$\mathcal{S} \setminus \widehat{\mathcal{S}}_h^\beta$	s_{sink}	1
s_{sink}	s_{sink}	1
Otherwise		0

I.e., the only change from $\widehat{P}$ is that P^1 uses the true dynamics from $\widehat{\mathcal{S}}^\beta$ to $\mathcal{S}$, while $\widehat{P}$ is the empirical mean.

We denote **occupancy measure** of this dynamics as $q_h^1(s \mid \pi) := q_h(s \mid \pi, P^1)$. Similar to $\widehat{q}$, it is over only states in $\mathcal{S}$, and excludes s_{sink}.

A.1.4 Shared dynamics Definitions and Notations

Remark A.9. At learning phase i, our algorithm (defined later) builds dynamics $\widehat{P}_i$. The occupancy measure at the beginning of phase i is $q_h(s \mid \pi, \{\widehat{P}_j\}_{j=0}^{i-1})$, for $h \leq i$. For simplicity, we write $q_h(s \mid \pi, \widehat{P})$, instead of $q_h(s \mid \pi, \{\widehat{P}_j\}_{j=0}^{h-1})$, when it is clear from the context.

Definition A.10. The states **transition matrix** from $\mathcal{S}$ to $\mathcal{S}$ at timestep h of dynamics P is $P_h^\pi[s, s'] = \sum_{a \in \mathcal{A}} P_h(s' \mid s, a)\pi_h(a \mid s)$.

The transition matrices for P_h, P_h^1, P_h^2 are $P_h^\pi, P_h^{1,\pi}, P_h^{2,\pi}$ accordingly.

Remark A.11. Note that the norm over the occupancy measures, $\|q_h(\cdot \mid \pi, \widehat{P})\|_1$ etc., is just over the states in $\mathcal{S}$, since we excluded s_{sink} from the $q_h(\cdot \mid \pi, \widehat{P}), q_h(\cdot \mid \pi, P^1), q^2$ vectors.

A.2 The Good Event

Definition A.12. The good event of timestep h, denoted by E_h, is the event in which simultaneously for every state $s \in \widehat{\mathcal{S}}_h^\beta$ and action $a \in \mathcal{A}$ it holds that

$$\|\widehat{P}_h(\cdot \mid s,a) - P_h^1(\cdot \mid s,a)\|_1 \leq \sqrt{\frac{\log(1/\delta') + 2S}{\frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)}}.$$

Definition A.13. We denote the good event up until h to be $E_{\leq h} = \cap_{i \leq h} E_i$. The good event is where every E_h happens, i.e., $E_{\leq H-1}$.

From now on, we will assume that $E_{\leq h}$ holds. Later we will show that it holds w.h.p. for every h, i.e., the good event holds w.h.p.

Definition A.14. We define δ' to be,

$$\delta' := \frac{\delta}{\text{supp}(m)} \leq \frac{\delta}{10S^2HA},$$

where $\text{supp}(m)$ is an upper bound on m. Note that δ' depends on m as well, but we can assume $m \leq (SHA/(\epsilon\delta))^3$ and we get $\log(1/\delta') = \log((SHA/(\epsilon\delta))^3) = 3\log(SHA/(\epsilon\delta))$ so it changes it by a constant factor only. See below Remark A.15 for more details.

Remark A.15. In the concentration bounds we use δ' and not directly δ, since the confidence intervals we use contain random variables. That said, we bound the number of agents m with a term that contains m. It works since what we means is that one should take a large enough m that satisfies it. Large enough m always exists, as roughly speaking we bound m from below with $\log(m)$ (i.e., we need to find m s.t. $m \gtrsim \log(m)$). Note that satisfying this condition changes the value only by constants, and it does not change the parameters' powers, i.e., the power of S, H, A and ϵ, or the $\log$ powers. One can see $\log(1/\delta')$ as of order of $\log(SHA/(\delta\epsilon))$.

A.3 Bounding $\widehat{P}$ with P^1 dynamics

Lemma A.16. *Let $s \in \widehat{\mathcal{S}}_h^\beta$. Assume that for each policy π*

$$\|q_h(\cdot \mid \pi, \widehat{P}) - q_h(\cdot \mid \pi, P^1)\|_1 \leq \frac{\beta}{3}.$$

Then,

$$\frac{2}{3}\beta \leq q_h(s \mid \widehat{\pi}^{h,s}, P^1).$$

Proof.

$$q_h(s \mid \widehat{\pi}^{h,s}, P^1) \geq q_h(s \mid \widehat{\pi}^{h,s}, \widehat{P}) - \frac{\beta}{3} \geq \beta - \frac{\beta}{3} = \frac{2}{3}\beta,$$

where the first inequality is from the assumption, and the second since $s \in \widehat{\mathcal{S}}_h^\beta$. □

Lemma A.17. *The bound on the occupancy measure multiplied by the confidence interval is bounded by α/S. Specifically, Assume $\beta/3 \leq q_h(s \mid \widehat{\pi}^{h,s}, P^1)$. Assume*

$$m \geq \max\{\frac{4S^3A(\log(1/\delta') + 2S)}{\alpha^2}, \frac{4HS^3A(\log(1/\delta') + 2S)}{\alpha}\}$$

Then,

$$(q_h(s \mid \widehat{\pi}^{h,s}, P^1) + \alpha h)\sqrt{\frac{\log(1/\delta') + 2S}{\frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)}} \leq \frac{\alpha}{S}$$

Proof. First, the term involving $q_h(s \mid \widehat{\pi}^{h,s}, P^1)$ satisfies,

$$\begin{aligned}
&q_h(s \mid \widehat{\pi}^{h,s}, P^1) \cdot \sqrt{\frac{\log(1/\delta') + 2S}{\frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)}} \\
&= \sqrt{\frac{SA(\log(1/\delta') + 2S)}{m}} \sqrt{q_h(s \mid \widehat{\pi}^{h,s}, P^1)} \\
&\leq \sqrt{\frac{SA(\log(1/\delta') + 2S)}{m}} \quad \text{(since } q_h(s \mid \widehat{\pi}^{h,s}, P^1) \leq 1\text{)} \\
&\leq \frac{\alpha}{2S},
\end{aligned}$$

And it holds from the assumption on a large enough m. Specifically,

$$\sqrt{\frac{SA(\log(1/\delta')+2S)}{m}} \leq \frac{\alpha}{2S}$$
$$\frac{SA(\log(1/\delta')+2S)}{m} \leq \frac{\alpha^2}{4S^2},$$

I.e.,

$$m \geq \frac{4S^3A(\log(1/\delta')+2S)}{\alpha^2}. \tag{6}$$

Next, the term involving αh is bounded as well,

$$\begin{aligned}
&\alpha h\sqrt{\frac{\log(1/\delta')+2S}{\frac{m}{SA}q_h(s \mid \widehat{\pi}^{h,s}, P^1)}} \\
&\leq \alpha h\sqrt{\frac{\log(1/\delta')+2S}{\frac{m}{SA}\cdot\frac{1}{3}\beta}} \quad \text{(from the assumption)} \\
&\leq \alpha H\sqrt{\frac{SA(\log(1/\delta')+2S)}{m}}\frac{1}{\sqrt{\frac{1}{3}3\alpha H}} \\
&= \sqrt{\frac{SA(\log(1/\delta')+2S)\alpha H}{m}} \\
&\leq \frac{\alpha}{2S},
\end{aligned}$$

where the last inequality is since we need the following equation to hold:

$$\sqrt{\frac{SA(\log(1/\delta')+2S)\alpha H}{m}} \leq \frac{\alpha}{2S}$$
$$\frac{SA(\log(1/\delta')+2S)\alpha H}{m} \leq \frac{\alpha^2}{4S^2}$$

And it holds from the assumption on a large enough m:

$$m \geq \frac{4HS^3A(\log(1/\delta')+2S)H}{\alpha}. \tag{7}$$

□

Definition A.18. Denote the bound on the number of agents with

$$m(S,A,H,\alpha,\delta') := \max\{\frac{4S^3A(\log(1/\delta')+2S)}{\alpha^2}, \frac{4HS^3A(\log(1/\delta')+2S)}{\alpha}\}.$$

Lemma A.19. *Assume we estimated P^1 until and include timestep h with `H-MARFE` algorithm, where the number of agents satisfies,*

$$m \geq m(S,A,H,\alpha,\delta').$$

Let π be a policy.

Assume that,

$$\|q_h(\cdot \mid \pi, \widehat{P}) - q_h(\cdot \mid \pi, P^1)\|_1 \leq \alpha h.$$

Then,

$$\|q_h(\cdot \mid \pi, \widehat{P})(\widehat{P}_h^{\pi} - P_h^{1,\pi})\|_1 \leq \alpha.$$

Proof. From the assumption, and since $\beta = 3\alpha H$, for every policy π' it holds that $|q_h(s \mid \pi', P^1) - q_h(s \mid \pi', \widehat{P})| \leq \beta/3$, Then the conditions of $Lemma\ A.16$ hold, and we get that for every $s \in \widehat{\mathcal{S}}_h^{\beta}$,

$$\frac{2}{3}\beta \leq q_h(s \mid \widehat{\pi}^{h,s}, P^1). \tag{8}$$

We now show that our confidence intervals are small enough.

Let $s \in \widehat{\mathcal{S}}_h^\beta$.

From the good event,

$$\|\widehat{P}_h(\cdot \mid s,a) - P_h^1(\cdot \mid s,a)\|_1 \le \sqrt{\frac{\log(1/\delta') + 2S}{\frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)}}. \tag{9}$$

Hence,

$$\begin{aligned}
&\sum_{s' \in \mathcal{S}} q_h(s \mid \pi, \widehat{P}) |\widehat{P}_h(s' \mid s, \pi_h(s)) - P_h^1(s' \mid s, \pi_h(s))| \\
&= q_h(s \mid \pi, \widehat{P}) \sum_{s' \in \mathcal{S}} |\widehat{P}_h(s' \mid s, \pi_h(s)) - P_h^1(s' \mid s, \pi_h(s))| \\
&\le q_h(s \mid \widehat{\pi}^{h,s}, \widehat{P}) \sum_{s' \in \mathcal{S}} |\widehat{P}_h(s' \mid s, \pi_h(s)) - P_h^1(s' \mid s, \pi_h(s))| \qquad \text{(by def. of } \widehat{\pi}^{h,s}) \\
&\le (q_h(s \mid \widehat{\pi}^{h,s}, P^1) + \alpha h) |\widehat{P}_h(s' \mid s, \pi_h(s)) - P_h^1(s' \mid s, \pi_h(s))| \quad (\text{since } \|\cdot\|_\infty \le \|\cdot\|_1) \\
&\le (q_h(s \mid \widehat{\pi}^{h,s}, P^1) + \alpha h) \sqrt{\frac{\log(1/\delta') + 2S}{\frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)}}. \quad \text{(from the good event, Equation (9))}
\end{aligned}$$

From Equation (8), the condition of $\beta/3 \le q_h(s \mid \widehat{\pi}^{h,s}, P^1)$ holds. Hence, we can use Lemma A.17, and we get,

$$(q_h(s \mid \widehat{\pi}^{h,s}, P^1) + \alpha h) \sqrt{\frac{\log(1/\delta') + 2S}{\frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)}} \le \frac{\alpha}{S}.$$

Therefore,

$$q_h(s \mid \pi, \widehat{P}) \sum_{s' \in \mathcal{S}} |\widehat{P}_h(s' \mid s, \pi_h(s)) - P_h^1(s' \mid s, \pi_h(s))| \le \frac{\alpha}{S}. \tag{10}$$

Summing over all $s' \in \mathcal{S}$ yields

$$\begin{aligned}
&\|\widehat{q}_h(\widehat{P}_h - P_h^1)\|_1 \\
&= \sum_{s' \in \mathcal{S}} \sum_{s \in \widehat{\mathcal{S}}_h^\beta} q_h(s \mid \pi, \widehat{P}) |\widehat{P}_h(s' \mid s, \pi_h(s)) - P_h^1(s' \mid s, \pi_h(s))| \\
&= \sum_{s \in \widehat{\mathcal{S}}_h^\beta} q_h(s \mid \pi, \widehat{P}) \sum_{s' \in \mathcal{S}} |\widehat{P}_h(s' \mid s, \pi_h(s)) - P_h^1(s' \mid s, \pi_h(s))| \\
&\le \sum_{s' \in \mathcal{S}} \frac{\alpha}{S} \quad \text{(from Equation (10))} \\
&\le \alpha.
\end{aligned}$$

Hence,

$$\|q_h(\cdot \mid \pi, \widehat{P})(\widehat{P}_h^\pi - P_h^{1,\pi})\|_1 \le \alpha,$$

and note that the L_1 norm is taken over the entire state space $\mathcal{S}$ and not just $\widehat{\mathcal{S}}_h^\beta$. □

Lemma A.20. *Assume we play* `H-MARFE` *algorithm, with m agents, where*

$$m \ge m(S, A, H, \alpha, \delta').$$

Assume that for every policy π,

$$\|q_h(\cdot \mid \pi, \widehat{P}) - q_h(\cdot \mid \pi, P^1)\|_1 \le \alpha h.$$

Then,

$$\|q_{h+1}(\cdot \mid \pi, \widehat{P}) - q_{h+1}(\cdot \mid \pi, P^1)\|_1 \le \alpha(h+1).$$

Proof. Since $E_{\leq h}$ holds:

$$\begin{aligned}
&\|\widehat{q}_{h+1} - q^1_{h+1}\|_1 \\
&= \|\widehat{q}_h \widehat{P}_h - q^1_h P^1_h\|_1 \\
&= \|\widehat{q}_h \widehat{P}_h - \widehat{q}_h P^1_h + \widehat{q}_h P^1_h - q^1_h P^1_h\|_1 \\
&\leq \|\widehat{q}_h \widehat{P}_h - \widehat{q}_h P^1_h\|_1 + \|\widehat{q}_h P^1_h - q^1_h P^1_h\|_1 \\
&= \|\widehat{q}_h (\widehat{P}_h - P^1_h)\|_1 + \|(\widehat{q}_h - q^1_h) P^1_h\|_1 \\
&\leq \|\widehat{q}_h (\widehat{P}_h - P^1_h)\|_1 + \|\widehat{q}_h - q^1_h\|_1 \quad \text{(from Lemma C.1)} \\
&\leq \|\widehat{q}_h (\widehat{P}_h - P^1_h)\|_1 + \alpha h \quad \text{(assumption)} \\
&\leq \alpha + \alpha h \quad \text{(from Lemma A.19)} \\
&\leq \alpha(h+1).
\end{aligned}$$

Recall we denote $\widehat{q}_h := q_h(s \mid \pi, \widehat{P})$, and similar for q^1_h, and we also omit the policy notation in the transition matrix $\widehat{P}_h := \widehat{P}^\pi_h$, and similar for P^1_h. See Remark A.9 for the notations and abbreviations. □

Corollary A.21. *As we assumed until now, we assume $E_{\leq h}$ holds (we explicitly assumed that for simplicity). Assume we play `H-MARFE` algorithm, with m agents, where*

$$m \geq m(S, A, H, \alpha, \delta').$$

Then,

$$\|q_{h+1}(\cdot \mid \pi, \widehat{P}) - q_{h+1}(\cdot \mid \pi, P^1)\|_1 \leq \alpha(h+1).$$

Proof. By induction.

For $h = 0$, trivial, since $\widehat{q}_0 = q^1_0$.

Assume true for h. For $h+1$ we use Lemma A.20. □

Lemma A.22. *Assume we play `H-MARFE` algorithm, with m agents, where*

$$m \geq m(S, A, H, \alpha, \delta').$$

Assume that for every policy π,

$$\|q_h(\cdot \mid \pi, \widehat{P}) - q_h(\cdot \mid \pi, P^1)\|_1 \leq \alpha h.$$

Then for every state in $\widehat{\mathcal{S}}^\beta_h$ and every action simultaneously

$$\mathbb{P}\left(N_h(s,a) \leq \frac{1}{2}\frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)\right) \leq \delta'/(2H).$$

Proof. From Lemma A.16, for $s \in \widehat{\mathcal{S}}^\beta_h$, $q_h(s \mid \widehat{\pi}^{h,s}, P^1) \geq \alpha H$.

$$\begin{aligned}
&\mathbb{P}\left(N_h(s,a) \leq \frac{1}{2}\frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)\right) \\
&\leq \mathbb{P}\left(N_h(s,a) \leq \frac{1}{2}\mathbb{E}\left(N_h(s,a)\right) \mid \mathcal{F}_{<h}\right) \\
&\leq e^{-\frac{1}{8} q_h(s \mid \widehat{\pi}^{h,s}, P^1) m/(SA)} \\
&\leq e^{-\frac{1}{8}\alpha H m/(SA)} \leq \frac{\delta'}{2HSA}.
\end{aligned}$$

$\mathcal{F}_{<h}$ is the filtration induced by what the agents saw up until the beginning of timestep h. I.e., $\mathcal{F}_{<0} = \Omega$.

The first inequality is since $\mathbb{E}\left(N_h(s,a) \mid \mathcal{F}_{<h}\right) \geq \frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)$ (each agent gets random state-action pair to explore).

The second inequality is Chernoff bound (Lemma C.6), and again $\mathbb{E}\left(N_h(s,a) \mid \mathcal{F}_{<h}\right) \geq \frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)$.

The last inequality is since $m \geq m(S,A,H,\alpha,\delta') \geq SA\log(2HSA/\delta)/(H\alpha)$.

From union bound over all states and actions, we get the bound. □

We denote with C_h the event that for every $s,a \in \widehat{\mathcal{S}}_h^\beta \times \mathcal{A}$ it holds that $N_h(s,a) > \frac{1}{2}\frac{m}{SA} q_h(s \mid \widehat{\pi}^{h,s}, P^1)$.

We now prove that the good event holds w.h.p., and hence will not assume that $E_{\leq h}$ holds, but rather will prove that it holds w.h.p. for every h.

Lemma A.23. *Assume we play* `H-MARFE` *algorithm with m agents, where*

$$m \geq \max\{m(S,A,H,\alpha,\delta')\}.$$

Then, with probability at least $1-\delta'$, the good event $E_{\leq H-1} = \bigcap_{h=0}^{H-1} E_h$ holds.

Proof. We prove by induction on h that

$$\mathbb{P}\left(E_{\leq h}\right) \geq 1 - (h+1)\delta'/H. \tag{11}$$

Base case ($h=0$). The event E_0 depends only on the trajectories collected in phase 0. Conditional on the (deterministic) initialization $\mathcal{F}_{<0}$: By Lemma A.22, C_h holds failure probability at most $\delta'/(2H)$.

By Lemma C.5 applied with $\mathcal{K} = N_0(s,a)$, the BHC concentration in terms of the realized $N_0(s,a)$ holds for each fixed (s,a) with failure probability at most $\delta'/(2HSA)$. A union bound over $(s,a) \in \widehat{\mathcal{S}}_0^\beta \times A$ gives total failure probability at most $\delta'/(2H)$. Since C_0 holds we can replace $N_0(s,a)$ with $\frac{1}{2}m/(SA)q_h(s \mid \widehat{\pi}^{h,s}, P^1)$. I.e.,

$$\|\widehat{P}_0(\cdot \mid s,a) - P_0^1(\cdot \mid s,a)\|_1 \leq \sqrt{\frac{\log(1/\delta') + 2S}{2N_0(s,a)}} \leq \sqrt{\frac{\log(1/\delta') + 2S}{\frac{m}{SA} q_0(s \mid \widehat{\pi}^{0,s}, P^1)}},$$

with probability at least $1-\delta'/(2SHA)$, and for every state (even though we have only one initial state, we keep the structure) and for every action it yields $1-\delta'/(2H)$.

A union bound over the two events yields $\mathbb{P}\left(E_0\right) \geq 1-\delta'/H$.

Inductive step. Suppose Equation (11) hold for h. We show it holds for $h+1$.

By the chain rule,

$$\mathbb{P}\left(E_{\leq h+1}\right) = \mathbb{P}\left(E_{\leq h}\right) \cdot \mathbb{P}\left(E_{h+1} \mid E_{\leq h}\right).$$

From Corollary A.21, since $E_{\leq h}$ holds, the hypothesis of Lemma A.22 is satisfied for $h+1$. Therefore, Lemma A.22 implies that C_{h+1} holds with probability at least $1-\delta'/(2H)$.

Similarly to the base case: Lemma C.5 applied with $\mathcal{K} = N_{h+1}(s,a)$ conditional on $\mathcal{F}_{<h+1}$ implies

$$\|\widehat{P}_{h+1}(\cdot \mid s,a) - P_{h+1}^1(\cdot \mid s,a)\|_1 \leq \sqrt{\frac{\log(1/\delta') + 2S}{2N_{h+1}(s,a)}} \leq \sqrt{\frac{\log(1/\delta') + 2S}{\frac{m}{SA} q_{h+1}(s \mid \widehat{\pi}^{h+1,s}, P^1)}},$$

with probability at least $1 - \frac{\delta'}{2SHA}$. Union bound across all states and actions yields that this happens with probability at least $1 - \frac{\delta'}{2H}$.

A union bound over this event and C_{h+1} yields $\mathbb{P}\left(E_{h+1} \mid E_{\leq h}\right) \geq 1-\delta'/H$. Hence

$$\mathbb{P}\left(E_{\leq h+1}\right) \geq \mathbb{P}\left(E_{\leq h}\right)(1-\delta'/H) \geq \mathbb{P}\left(E_{\leq h}\right) - \delta'/H \geq 1-(h+2)\delta'/H,$$

proving the induction step.

The claim is true in particular for $h = H-1$, $\mathbb{P}\left(E_{\leq H-1}\right) \geq 1-\delta'$. □

Corollary A.24. *Assume the good event holds. Assume we play* `H-MARFE` *algorithm, with m agents, where*

$$m \geq m(S, A, H, \alpha, \delta').$$

Then, for every timestep h, for every policy π,

$$\|q_h(\cdot \mid \pi, \widehat{P}) - q_h(\cdot \mid \pi, P^1)\|_1 \leq \alpha h.$$

Proof. Immediate from the definition of the good event $E_{\leq H-1}$ (Definition A.13) and from Corollary A.21. □

Theorem A.25. *For any policy and reward function, the value functions of the estimated dynamics $\widehat{P}$ and the dynamics P^1 are close. Specifically:*

Let $\epsilon > 0, \delta > 0$. Assume we play `H-MARFE`*, with H phases, and the number of agents is at least,*

$$m \geq \max\{\frac{4S^3A(\log(1/\delta') + 2S)}{\alpha^2}, \frac{4HS^3A(\log(1/\delta') + 2S)}{\alpha}\}.$$

Then, with probability higher than $1 - \delta$ the following holds. For every policy π, for every reward function $r : H \times S \times A \to [0, 1]$,

$$|V^{\pi}_{P^1,r} - V^{\pi}_{\widehat{P},r}| \leq \frac{1}{2}\alpha H^2.$$

Also, if we also assume that $\alpha \leq \frac{2\epsilon}{H^2}$, then,

$$|V^{\pi}_{P^1,r} - V^{\pi}_{\widehat{P},r}| \leq \epsilon.$$

Proof. The good event holds with probability higher than $1 - \delta$, see Lemma A.23. Conditioning on the good event, the following holds.

From Corollary A.24, for every $h \in [H]$,

$$\|q_h^{1,\pi} - \widehat{q}_h^{\pi}\|_1 \leq \alpha h,$$

where the norm is over all states in $\mathcal{S}$.

For the value function we get,

$$\begin{aligned} V^{\pi}_{\widehat{P},r} &= \mathbb{E}_{\widehat{P},\pi}(\sum_{h=0}^{H-1}(r_h(s_h, a_h))) \\ &= \sum_{h=0}^{H-1} \mathbb{E}_{\widehat{P}} \left(\sum_{a \in \mathcal{A}} r_h(s_h, a)\pi_h(a \mid s_h) \right) \\ &:= \sum_{h=0}^{H-1} \mathbb{E}_{\widehat{P}}(f^{\pi}(s_h)) \end{aligned}$$

Where the last equality is for simplicity of notations, as we denote the expectation of the reward on s_h with $f^{\pi}(s_h)$,

$$f^{\pi}(s_h) := \sum_{a \in \mathcal{A}} r_h(s_h, a)\pi_h(a \mid s_h).$$

Note that $0 \leq f^{\pi}(s) \leq 1$.

Similarly for $q_h(\cdot \mid \pi, P^1)$. Hence, from Lemma C.7,

$$|\mathbb{E}_{\widehat{P}}(f^{\pi}(s_h)) - \mathbb{E}_{P^1}(f^{\pi}(s_h))| \leq \|\widehat{q}_h - q_h^1\|_1.$$

Therefore,

$$
\begin{aligned}
&|V^{\pi}_{\widehat{P},r} - V^{\pi}_{P^1,r}| \\
&\leq \sum_{h=0}^{H-1} |\mathbb{E}_{\widehat{P}}(f^{\pi}(s_h)) - \mathbb{E}_{P^1}(f^{\pi}(s_h))| \\
&\leq \sum_{h=0}^{H-1} 1 \cdot \|\widehat{q}_h - q^1_h\|_1 \\
&\leq \sum_{h=0}^{H-1} \alpha h \quad \text{(from Corollary A.24)} \\
&= \alpha \frac{H(H-1)}{2} \\
&\leq \frac{1}{2}\alpha H^2.
\end{aligned}
$$

Since $\alpha \leq \frac{2\epsilon}{H^2}$ we get,

$$\frac{1}{2}\alpha H^2 \leq \epsilon.$$

□

A.4 All four dynamics have similar value functions

A.4.1 The P^2 dynamics

Analogous to the P^1 dynamics, we define the P^2 dynamics. This transition function serves as a theoretical bridge between the true dynamics P and the P^1 dynamics P^1. While P^1 is constructed based on empirical reachability, P^2 is defined using the true transition dynamics but restricted to states that remain significantly reachable under the inductive constraint.

Specifically, in P^2, transitions continue only from states that are truly β-reachable within this restricted model. Unlike $\widehat{\mathcal{S}}^{\beta}_h$, which depends on the collected data, the set of reachable states in this dynamics is deterministic.

Let $q_h(s \mid \pi, P^2) := q^2_h(s \mid \pi)$ be the **occupancy measure** of P^2, excluding s_{sink}. Note, it is not q-square, but just a notation.

Definition A.26. We define the set of inductive β-significant states at timestep h as:

$$\mathcal{S}^2_h := \{s \in \mathcal{S} \mid \max_{\pi} q_h(s \mid \pi, P^2) \geq 2\beta\}.$$

Note that $\mathcal{S}^2_h$ is a deterministic set.

Transition function for P^2. As before, we introduce a sink state s_{sink} to collect the probability mass of all non-reachable states. The state space of P^2 is $\mathcal{S} \cup \{s_{\text{sink}}\}$, and the transition kernel P^2_h is defined as follows:

s	s'	$P^2_h(s' \mid s, a)$
$\mathcal{S}^2_h$	$\mathcal{S}$	$P_h(s' \mid s, a)$
$\mathcal{S} \setminus \mathcal{S}^2_h$	s_{sink}	1
s_{sink}	s_{sink}	1
Otherwise		0

By this definition, if a state s is not β-reachable in P^2 (i.e., $s \notin \mathcal{S}^2_h$), it acts as a termination point where all subsequent transitions lead to s_{sink} with probability 1, effectively pruning the dynamics.

Lemma A.27. *If $\mathcal{S}_i^2 \subseteq \widehat{\mathcal{S}}_i^\beta$ for every $i < h$, then for every policy π and state $s \in \mathcal{S}_h^2$, $q_h(s \mid \pi, P^2) \leq q_h(s \mid \pi, P^1)$.*

Proof. By induction. Immediate for $h = 0$. Assume true for h. For $h+1$: Assume $\mathcal{S}_i^2 \subseteq \widehat{\mathcal{S}}_i^\beta$ for every $i < h+1$. Let $s \in \widehat{\mathcal{S}}_{h+1}^\beta$. Denote $q_h^2(s \mid \pi) := q_h(s \mid \pi, P^2)$ and $q_h^1(s \mid \pi) = q_h(s \mid \pi, P^1)$.

$$\begin{aligned}
q_{h+1}^1(s \mid \pi) &= \sum_{s' \in \widehat{\mathcal{S}}_h^\beta} q_h^1(s' \mid \pi) P_h^1(s \mid s', \pi_h(s')) \\
&= \sum_{s' \in \widehat{\mathcal{S}}_h^\beta} q_h^1(s' \mid \pi) P_h(s \mid s', \pi_h(s')) \\
&\geq \sum_{s' \in \mathcal{S}_h^2} q_h^1(s' \mid \pi) P_h(s \mid s', \pi_h(s')) \quad \text{(since } \mathcal{S}_h^2 \subseteq \widehat{\mathcal{S}}_h^\beta\text{)} \\
&\geq \sum_{s' \in \mathcal{S}_h^2} q_h^2(s' \mid \pi) P_h(s \mid s', \pi_h(s')) \quad \text{(induction hypothesis)} \\
&= q_{h+1}^2(s \mid \pi)
\end{aligned}$$

□

Lemma A.28. *Assume*

$$m \geq m(S, A, H, \alpha, \delta').$$

Then for every phase $h \in \{0, \ldots, H\}$, it holds that $\mathcal{S}_h^2 \subseteq \widehat{\mathcal{S}}_h^\beta$.

Proof. Denote $q_h^2(s \mid \pi) := q_h(s \mid \pi, P^2)$ and $q_h^1(s \mid \pi) = q_h(s \mid \pi, P^1)$.

By induction. $h = 0$ true since $\mathcal{S}_0^2 = \widehat{\mathcal{S}}_0^\beta = \{s_0\}$. Assume true for every j s.t. $j \leq h$. For $h+1$: Let $s \in \mathcal{S}_{h+1}^2$. From the definition, there exists π s.t.

$$2\beta \leq q_{h+1}^2(s \mid \pi).$$

Hence, from the induction hypothesis and from Lemma A.27,

$$q_{h+1}^2(s \mid \pi) = \sum_{s' \in \mathcal{S}_h^2} q_h^2(s' \mid \pi) P_h(s \mid s', \pi) \leq \sum_{s' \in \widehat{\mathcal{S}}_h^\beta} q_h^2(s' \mid \pi) P_h(s \mid s', \pi) \leq \sum_{s' \in \widehat{\mathcal{S}}_h^\beta} q_h^1(s' \mid \pi) P_h(s \mid s', \pi),$$

where the first inequality is from the induction hypothesis and the second from Lemma A.27.

In each timestep, from Corollary A.24, we still have good estimation for all states, whether they are in $\widehat{\mathcal{S}}_h^\beta$ or not. Therefore,

$$\sum_{s' \in \widehat{\mathcal{S}}_h^\beta} q_h^1(s' \mid \pi) P_h(s \mid s', \pi) = q_{h+1}^1(s \mid \pi) \leq \widehat{q}_{h+1}(s \mid \pi) + \frac{\beta}{3} \leq \widehat{q}_{h+1}(s \mid \widehat{\pi}^{h,s}) + \frac{\beta}{3},$$

where the last inequality is by the definition of $\widehat{\pi}^{h,s}$.

Putting all the pieces together,

$$\widehat{q}_{h+1}(s \mid \widehat{\pi}^{h,s}) + \frac{\beta}{3} \geq q_{h+1}^2(s \mid \pi) \geq 2\beta.$$

Hence,

$$\widehat{q}_{h+1}(s \mid \widehat{\pi}^{h,s}) \geq 1\frac{2}{3}\beta \geq \beta,$$

and $s \in \widehat{\mathcal{S}}_{h+1}^\beta$. □

Corollary A.29. *Assume*

$$m \geq m(S, A, H, \alpha, \delta').$$

For every policy π, For every reward function $r : H \times S \times A \to [0, 1]$, it holds that,

$$V^{\pi}(s_0 \mid P^1, r) \geq V^{\pi}(s_0 \mid P^2, r).$$

Proof. From Lemma A.28 and Lemma A.27,

$$\begin{aligned}
V^{\pi}(s_0 \mid P^1, r) &= \sum_{h=0}^{H} \sum_{s \in \widehat{\mathcal{S}}_h^{\beta}} \sum_{a \in \mathcal{A}} q_h(s \mid \pi, P^1) r_h(s, a) \cdot \pi_h(a \mid s) \\
&\geq \sum_{h=0}^{H} \sum_{s \in \mathcal{S}_h^2} \sum_{a \in \mathcal{A}} q_h(s \mid \pi, P^1) r_h(s, a) \cdot \pi_h(a \mid s) \\
&\geq \sum_{h=0}^{H} \sum_{s \in \mathcal{S}_h^2} \sum_{a \in \mathcal{A}} q_h(s \mid \pi, P^2) r_h(s, a) \cdot \pi_h(a \mid s) \\
&= V^{\pi}(s_0 \mid P^2, r),
\end{aligned}$$

where the first inequality is from Lemma A.28 and the second from Lemma A.27. □

Lemma A.30. *For every policy π, For every reward function $r : H \times S \times A \to [0, 1]$, it holds that $V_h^{\pi}(s \mid P, r) \geq V_h^{\pi}(s \mid P^1, r)$.*

Proof. Similar to the proof Corollary A.29, but here we use the definitions directly and we do not need the lemmas. Here we have $\widehat{\mathcal{S}}_h^{\beta} \subseteq \mathcal{S}$, and since P^1 is just a truncation of P it holds that $q_h(s \mid \pi, P^1) \leq q_h(s \mid \pi, P)$.

$$\begin{aligned}
V^{\pi}(s_0 \mid P, r) &= \sum_{h=0}^{H} \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} q_h(s \mid \pi, P) r_h(s, a) \cdot \pi_h(a \mid s) \\
&\geq \sum_{h=0}^{H} \sum_{s \in \widehat{\mathcal{S}}_h^{\beta}} \sum_{a \in \mathcal{A}} q_h(s \mid \pi, P^1) r_h(s, a) \cdot \pi_h(a \mid s) \\
&= V^{\pi}(s_0 \mid P^1, r).
\end{aligned}$$

□

Lemma A.31. *The value function under the real dynamics P is bounded from above by the value function under P^2 and an expression that depends on β.*

Specifically, For every policy π, For every reward function $r : H \times S \times A \to [0, 1]$,

$$V^{\pi}(s_0 \mid P, r) \leq V^{\pi}(s_0 \mid P^2, r) + 2\beta H^2 S.$$

Proof. Denote with $G^{2\beta}$ the event in which we were only at P^2. I.e., in every timestep h we visit only states that are in $\mathcal{S}_h^2$.

From the definition of $\mathcal{S}^2$, in every timestep h, we remove at most $S2\beta$ probability mass. Hence,

$$\Pr(\Omega \setminus G^{2\beta}) \leq HS2\beta. \tag{12}$$

Therefore,

$$\Pr(\Omega \setminus G^{2\beta}) \mathbb{E}\left(\sum_h r_h(s_h, \pi_h(s_h)) \mid \Omega \setminus G^{2\beta}\right) \leq \Pr(\Omega \setminus G^{2\beta}) H \leq H^2 S 2\beta.$$

where the first inequality is since the rewards are bounded by 1, and the second is from $Equation$ (12).

Putting both cases together, we get:

$$V^{\pi}(s_0 \mid P, r)$$
$$\mathbb{E}\left(\sum_h r_h(s_h, a_h) \mid G^{2\beta}\right)\mathbb{P}\left(G^{2\beta}\right) + \mathbb{E}\left(\sum_h r_h(s_h, a_h) \mid \Omega \setminus G^{2\beta}\right)\mathbb{P}\left(\Omega \setminus G^{2\beta}\right)$$
$$V^{\pi}(s_0 \mid P^2, r)\mathbb{P}\left(G^{2\beta}\right) + H^2 S \cdot 2\beta.$$

□

Corollary A.32. *The dynamics P^1 and the real dynamics P have similar value functions. Formally, assume*

$$m \geq m(S, A, H, \alpha, \delta').$$

Then,

$$V^{\pi}(s_0 \mid P^1, r) \leq V^{\pi}(s_0 \mid P, r) \leq V^{\pi}(s_0 \mid P^1, r) + H^2 S 2\beta.$$

And for $\alpha \leq \epsilon/(6H^3S)$, we get,

$$V^{\pi}(s_0 \mid P^1, r) \leq V^{\pi}(s_0 \mid P, r) \leq V^{\pi}(s_0 \mid P^1, r) + \epsilon.$$

Proof. We bound the value function of P^1 with P, $V^{\pi}(s_0 \mid P^1, r) \leq V^{\pi}(s_0 \mid P, r)$, from Lemma A.30. From Lemma A.31 we can bound the value function of P with P^2,

$$V^{\pi}(s_0 \mid P, r) \leq V^{\pi}(s_0 \mid P^2, r) + H^2 S 2\beta.$$

From Corollary A.29 we bound the value function of P^2 with P^1,

$$V^{\pi}(s_0 \mid P^2, r) + H^2 S 2\beta \leq V^{\pi}(s_0 \mid P^1, r) + H^2 S 2\beta.$$

Together, we have the first result. For $\alpha \leq \epsilon/(6H^3S)$ we get,

$$H^2 S 2\beta = H^2 S 2 \cdot 3\alpha H \leq \epsilon.$$

□

Lemma A.33. *Assume*

$$m \geq \max\{\frac{4S^3A(\log(1/\delta') + 2S)}{\alpha^2}, \frac{4HS^3A(\log(1/\delta') + 2S)}{\alpha}\},$$

then Corollary A.32 and Theorem A.25 hold.

Proof. Theorem A.25 requires

$$\alpha \leq \frac{2\epsilon}{H^2}.$$

Corollary A.32 requires

$$\alpha \leq \epsilon/(6H^3S).$$

Both requirements on α hold when

$$\alpha \leq \frac{\epsilon}{6H^3S}.$$

Lemma A.17 requires

$$m \geq m(S, A, H, \alpha, \delta') := \max\{\frac{4S^3A(\log(1/\delta') + 2S)}{\alpha^2}, \frac{4HS^3A(\log(1/\delta') + 2S)}{\alpha}\}.$$

Hence, all the requirements are fulfilled when,

$$m \geq \max\{\frac{4S^3A(\log(1/\delta') + 2S)}{(\frac{\epsilon}{6H^3S})^2}, \frac{4HS^3A(\log(1/\delta') + 2S)}{\frac{\epsilon}{6H^3S}}\}$$
$$= \max\{144\frac{S^5H^6A(\log(1/\delta') + 2S)}{\epsilon^2}, 24\frac{S^4H^4A(\log(1/\delta') + 2S)}{\epsilon}\},$$

I.e.,

$$m \geq 144\frac{S^5H^6A(\log(1/\delta') + 2S)}{\epsilon^2}. \tag{13}$$

□

Corollary A.34. *The dynamics $\widehat{P}$ and P are close. Specifically: Let $\epsilon > 0, \delta > 0$. Assume we play* H-MARFE*, with H phases, and*

$$m \geq 144 \frac{S^5 H^6 A(\log(1/\delta') + 2S)}{\epsilon^2}.$$

Then, with probability higher than $1 - \delta$, for every stochastic Markovian policy π, for every reward function $r : H \times S \times A \to [0, 1]$, for every dynamics P with horizon H,

$$|V^{\pi}_{P,r} - V^{\pi}_{\widehat{P},r}| \leq 2\epsilon$$

Proof. From Lemma A.33, m is large enough, hence, Corollary A.32, and Theorem A.25 hold. Therefore,

$$\begin{aligned}
&|V^{\pi}_{P,r} - V^{\pi}_{\widehat{P},r}| \\
&= |V^{\pi}_{P,r} - V^{\pi}_{P^1,r} + V^{\pi}_{P^1,r} - V^{\pi}_{\widehat{P},r}| \\
&\leq |V^{\pi}_{P,r} - V^{\pi}_{P^1,r}| + |V^{\pi}_{P^1,r} - V^{\pi}_{\widehat{P},r}| \\
&\leq \epsilon + |V^{\pi}_{P^1,r} - V^{\pi}_{\widehat{P},r}| \\
&\leq 2\epsilon.
\end{aligned}$$

Where the second inequality follows from Corollary A.32, and the last from Theorem A.25. □

Lemma A.35. *Assume there are two dynamics P, P', s.t. with probability at least $1 - \delta$ for every reward function r and policy π it holds that $|V^{\pi}_{P,r} - V^{\pi}_{P',r}| \leq \epsilon$.*

Then the following holds: With probability higher than $1 - \delta$, for every reward function r,

$$V^{\star}_{P,r} - V^{\pi'^{\star}}_{P,r} \leq 2\epsilon,$$

where $\pi'^{\star} \in \arg\max_{\pi} V^{\pi}_{P',r}$.

Proof.

$$\begin{aligned}
&V^{\star}_{P,r} - V^{\pi}_{P,r} \\
&= V^{\pi^{\star}}_{P,r} - V^{\pi^{\star}}_{P',r} + V^{\pi^{\star}}_{P',r} - V^{\pi'^{\star}}_{P',r} + V^{\pi'^{\star}}_{P',r} - V^{\pi}_{P',r} + V^{\pi}_{P',r} - V^{\pi}_{P,r} \\
&\leq \epsilon + V^{\pi^{\star}}_{P',r} - V^{\pi'^{\star}}_{P',r} + \epsilon \\
&\leq 2\epsilon,
\end{aligned}$$

where the first inequality is from the assumption and the second is since $\pi'^{\star}$ is optimal in P'. □

Theorem A.36. *We can learn with only H phases. Specifically: Let $\epsilon > 0, \delta > 0$. Assume we play* H-MARFE*, with H phases, and,*

$$m \geq 2304 \frac{S^5 H^6 A(\log(1/\delta') + 2S)}{\epsilon^2}.$$

Then, with probability higher than $1 - \delta$, for every reward function r,

$$V^{\star}_{P,r} - V^{\widehat{\pi}_r}_{P,r} \leq \epsilon,$$

where $\widehat{\pi}_r \in \arg\max_{\pi} V^{\pi}_{\widehat{P},r}$.

Proof. In Corollary A.34, we set the error to be $\epsilon/4$. Then we use Lemma A.35 with error of $\epsilon/2$ and the proof follows. □

B Lower Bound

In this chapter we prove our lower bound results.

Remark B.1. Here we assume that, after the learning phases are completed, the algorithm does not directly output the estimated dynamics $\widehat{P}$, but instead later receives a reward function and is required to return a policy. This generalizes the model considered in the main text.

Moreover, our policy is stochastic and, without loss of generality, we assume the reward function is deterministic, as in our upper bound; see Remark A.5 and Remark A.4.

B.1 The Key-Dynamic Framework

The Key-Dynamic is characterized by a minimal state space and deterministic transitions that simulate a "lock" that only one sequence of actions can open.

The **state space** $\mathcal{S}$ consists of only two states:

$$\mathcal{S} = \{s_{\text{sink}}, s^\star\}.$$

The **initial state** is always the "informative" state:

$$s_0 = s^\star.$$

The transition functions P_h are deterministic. In state $s^\star$, there exists exactly one "correct" action $a_h^\star$ at each timestep h that remains the agent in $s^\star$. Any other action leads to the absorbing state s_{sink}. Once in s_{sink}, the agent remains there regardless of the actions chosen.

The **transition function** is defined as follows:

s	s'	$P_h(s' \mid s, a)$
$s^\star$	$s^\star$	$\mathbb{1}(a = a_h^\star)$
$s^\star$	s_{sink}	$\mathbb{1}(a \neq a_h^\star)$
s_{sink}	$s^\star$	0
s_{sink}	s_{sink}	1

Definition B.2. We call the sequence of actions that keep the agent in $s^\star$ the **key** of the dynamics. We also call these actions the **good actions**.

Although we explore without any reward, the interesting reward function will be the reward that tells if the algorithm knows the "key" of the Key-Dynamic.

Definition B.3. We denote with r^{key} the following reward function.

$$r_h^{\text{key}}(s, a) = \mathbb{1}(h = H - 1, s = s^\star, a = a_{H-1}^\star).$$

B.1.1 Structural Implications

Below are several structural implications.

- **Action Sequences:** There are A^H possible action sequences, but only one unique sequence $(a_1^\star, a_2^\star, \ldots, a_H^\star)$ maintains the agent in $s^\star$.
- **Information Gain:** If an agent makes a mistake at any step h, it transitions to s_{sink} for the remainder of the episode. In this state, the agent receives no new information regarding the correct actions for steps $h + 1 \ldots H$.
- **Visual Analogy:** The dynamics can be visualized as two parallel lines of length H. The agent traverses the $s^\star$ line as long as it provides the "correct key" (actions). A single error drops the agent into the s_{sink} line, where it remains until the end of the horizon.
- **dynamics Space:** There are A^H distinct Key-Dynamic, each corresponding to a unique sequence of optimal actions.

B.2 The Key-Dynamic is learnable in one phase with $m = A^H$

Proposition B.4. *When the number of agents is $m = A^H$, each such Key-Dynamic is learnable in one phase. I.e., Assume $m = A^H$. There exists an algorithm s.t. for every $\epsilon > 0, \delta > 0$, a Key-Dynamic P^{key}, and a reward function $r : H \times S \times A \to [0, 1]$, after 1 phase of learning it holds that,*

$$\Pr\left(V^\star(s_0; P^{\text{key}}, r) - V^{\widehat{\pi}}(s_0; P^{\text{key}}, r) \leq \epsilon\right) \geq 1 - \delta.$$

Furthermore, the algorithm is deterministic and it always outputs the best policy, i.e., $\widehat{\pi} = \pi^\star$.

Proof. We construct a deterministic algorithm that will output the best policy. Let us enumerate all the possible action selections in each timestep, regardless of the states. We have A^H such selections, since we have A actions for each timestep. Let us enumerate the agents. The algorithm sends the agent of index i to action selections in index i, regardless of the state the agent reaches.

Since we send an agent to each action selection, one agent is sent to the selection we denote with $i^\star$, the selection that keeps in $s^\star$: $(a_0^\star, a_1^\star, \ldots, a_{H-1}^\star)$.

Since we restrict the learning to Key-Dynamic, all agents start at $s^\star$, and one agent, the one in index $i^\star$, stays in $s^\star$. Therefore, we know the sequence of the actions that keep in $s^\star$. Hence, we learned completely the transition function of the dynamics we explored, P^{key}. The transition function we estimated is as follows,

s	s'	$\widehat{P}_h(s' \mid s, a)$
$s^\star$	$s^\star$	$\mathbb{1}(a = a_h^\star)$
$s^\star$	s_{sink}	$\mathbb{1}(a \neq a_h^\star)$
s_{sink}	$s^\star$	0
s_{sink}	s_{sink}	1

and $\widehat{P} = P^{\text{key}}$.

Since the algorithm and the dynamics are deterministic, we get that for any reward function r, the algorithm outputs $\widehat{\pi} = \pi^\star$. Hence,

$$\Pr\left(V^\star(s_0; P^{\text{key}}, r) - V^{\widehat{\pi}}(s_0; P^{\text{key}}, r) = 0\right) = 1,$$

and we managed to learn the Key-Dynamic in one phase. □

B.3 The Expectation of Discovering the Key Drops Quickly

Definition B.5. The good agents, denoted with $\mathcal{G}_h^i(P^{\text{key}})$, are the group of agents that arrive at state $s^\star$ at timestep h at dynamics P^{key} with a fixed algorithm $\mathtt{A}$ at phase i. For the ease of notations, we drop the dynamics notations where they are clear from the context.

Definition B.6. When the problem is a single-phase learning, we denote the group of good agents $\mathcal{G}_h$, without the phase notations.

Lemma B.7. *For every algorithm $\mathtt{A}$ with one learning phase, For any Key-Dynamic with horizon H, P_H^{key}, there exists a Key-Dynamic of horizon $H + 1$, denoted with P_{H+1}^{key}, that has the same prefix of length H as in P_H^{key}, s.t. the expected number of the good agents on the dynamics P_{H+1}^{key} drops by $1/A$. I.e.,*

$$\mathbb{E}\left(|\mathcal{G}_{H+1}|\right) \leq \frac{\mathbb{E}\left(|\mathcal{G}_H|\right)}{A},$$

where $\mathbb{E}\left(|\mathcal{G}_H|\right) = \mathbb{E}\left(|\mathcal{G}_H(P_H^{\text{key}})|\right) = \mathbb{E}\left(|\mathcal{G}_H(P_{H+1}^{\text{key}})|\right)$.

Proof. Denote with α_i the action choice of agent i at timestep H (the transition to the last timestep). When the action in the key at the timestep H is a, we get,

$$\mathbb{E}\left(|\mathcal{G}_{H+1}| \mid \mathcal{G}_H\right) = \mathbb{E}\left(\sum_{i \in \mathcal{G}_H} \mathbb{1}(\alpha_i = a \mid \mathcal{G}_H)\right) = \sum_{i \in \mathcal{G}_H} \mathbb{P}\left(\alpha_i = a \mid \mathcal{G}_H\right).$$

We use the probabilistic method. We define a uniform distribution over the actions. More formally, the distribution is over the dynamics of horizon $H+1$, with their prefix the same as P_H^{key}. there are A such dynamics. The expectation over this distribution is denoted with $\mathbb{E}_{a\sim\mathcal{A}}$.

$$\begin{aligned}
&\mathbb{E}_{a\sim\mathcal{A}}\left(\sum_{i\in\mathcal{G}_H}\mathbb{P}\left(\alpha_i=a\mid\mathcal{G}_H\right)\right)\\
&=\frac{1}{A}\sum_{a\in\mathcal{A}}\sum_{i\in\mathcal{G}_H}\mathbb{P}\left(\alpha_i=a\mid\mathcal{G}_H\right)\\
&=\frac{1}{A}\sum_{i\in\mathcal{G}_H}\sum_{a\in\mathcal{A}}\mathbb{P}\left(\alpha_i=a\mid\mathcal{G}_H\right)\\
&=\frac{1}{A}\sum_{i\in\mathcal{G}_H}1\quad\text{(since every agent chooses one action)}\\
&=\frac{|\mathcal{G}_H|}{A}.
\end{aligned}$$

Together we get,

$$\begin{aligned}
&\mathbb{E}_{a\sim\mathcal{A}}\left(\mathbb{E}\left(|\mathcal{G}_{H+1}|\right)\right)\\
&=\mathbb{E}_{a\sim\mathcal{A}}\left(\mathbb{E}\left(\mathbb{E}\left(|\mathcal{G}_{H+1}|\mid\mathcal{G}_H\right)\right)\right)\\
&=\mathbb{E}_{a\sim\mathcal{A}}\left(\mathbb{E}\left(\sum_{i\in\mathcal{G}_H}\mathbb{P}\left(\alpha_i=a\mid\mathcal{G}_H\right)\right)\right)\\
&=\mathbb{E}\left(\mathbb{E}_{a\sim\mathcal{A}}\left(\sum_{i\in\mathcal{G}_H}\mathbb{P}\left(\alpha_i=a\mid\mathcal{G}_H\right)\right)\right)\\
&=\mathbb{E}\left(\frac{|\mathcal{G}_H|}{A}\right).
\end{aligned}$$

Then, there exists an dynamics of horizon $H+1$, denoted with P_{H+1}^{key}, with its prefix as P_H^{key}, and the next action in the key is $a^\star\in\mathcal{A}$, s.t.,

$$\mathbb{E}\left(|\mathcal{G}_{H+1}(P_{H+1}^{\text{key}})|\right)\le\frac{\mathbb{E}\left(|\mathcal{G}_H(P_{H+1}^{\text{key}})|\right)}{A}.$$

It follows from the prefixes equality that $\mathcal{G}_H(P_{H+1}^{\text{key}})=\mathcal{G}_H(P_H^{\text{key}})$. □

Corollary B.8. *For every algorithm $\mathtt{A}$ with one learning phase, for every horizon H, there exists a Key-Dynamic with horizon H s.t. for any $h\le H$ and $i\le h$,*

$$\mathbb{E}\left(|\mathcal{G}_h|\right)\le\frac{\mathbb{E}\left(|\mathcal{G}_i|\right)}{A^{h-i}}.$$

Proof. By induction on H. For $H=0$ it is true by definition. Assume true for H and call this dynamics P_H^{key}. For $H+1$: From Lemma B.7, there exists an dynamics P_{H+1}^{key} of horizon $H+1$ that satisfies $\mathbb{E}\left(|\mathcal{G}_{H+1}|\right)\le\mathbb{E}\left(|\mathcal{G}_H|/A\right)$, and its prefix is P_H^{key}. Therefore, from the prefixes equality, for every $h\le H$ and $i\le h$ it holds that $\mathbb{E}\left(|\mathcal{G}_h|\right)\le\frac{\mathbb{E}(|\mathcal{G}_i|)}{A^{h-i}}$. For $h=H+1$ it follows since for every $i\le H+1$,

$$\mathbb{E}\left(|\mathcal{G}_{H+1}|\right)\le\frac{\mathbb{E}\left(|\mathcal{G}_H|\right)}{A}\le\frac{\mathbb{E}\left(|\mathcal{G}_i|\right)}{A^{H-i}}\cdot\frac{1}{A}\le\frac{\mathbb{E}\left(|\mathcal{G}_i|\right)}{A^{H+1-i}},$$

where the second inequality is from the induction hypothesis. The dynamics that was constructed is indeed a Key-Dynamic. □

B.4 Lower Bound for Multiple Phases

In this section the algorithm has ρ phases, where $\rho < H$. Similar to one phase of learning, we construct a dynamics s.t. the expected number of good agents drops by a factor of $1/A$ in every timestep. Here we use the r^{key}, which indicates if the algorithm discovers the key or not. We will have a high probability that the algorithm does not discover the entire key, and it follows that with high probability the algorithm has high error.

Lemma B.9. *The algorithm cannot do better than just guessing the rest of the key. Formally, In one phase learning problem, for any algorithm $\mathtt{A}$, for any sequence of $H - H'$ actions $\bar{a}_{H-H'}$, there exists a Key-Dynamic of horizon H, denoted by P^{key}, with the first $H - H'$ good actions as $\bar{a}_{H-H'}$, such that if there are 0 agents at $s^\star$ at $h = H'$, then the algorithm will have $1/A^{H'}$ error at expectation. I.e.,*

$$\mathbb{E}\left(V^{\widehat{\pi}}(s_0; P^{\text{key}}, r^{\text{key}}) \mid |\mathcal{G}_{H'}| = 0\right) \leq \frac{1}{A^{H'}},$$

where $|\mathcal{G}_{H'}| = 0$ is the event that there are 0 agents at $s^\star$ at $h = H'$ at the learning phase, and $\mathbb{E}$ means the expectation over the algorithm choices.

Proof. Here we use the Probabilistic Method. We will use the fact that when the expectation of a random variable X has the value of $\mathbb{E}(X)$, there exists a point in the probability space $\omega \in \Omega$ s.t. $X(\omega) \leq \mathbb{E}(X)$.

All the Key-Dynamic differ only by the actions that keep the agent in $s^\star$. Let us fix the first $H - H'$ of the actions. So, for these actions we have $A^{H'}$ different Key-Dynamic with horizon of H. Let us enumerate them with $P^{\text{key,i}}$.

Each algorithm has an interaction stage, i.e., the stage in which the agents explore the dynamics. After this stage, the algorithm will decide on the policy, according to the reward function it gets. Let $\mathcal{I}_{\mathtt{A}}(P^{\text{key,j}})$ be the interaction of algorithm $\mathtt{A}$ with dynamics $P^{\text{key,j}}$.

If there are zero agents at the informative state $s^\star$ at timestep H', the interactions are the same for every Key-Dynamic with the same prefix of H' actions. I.e.,

$$\mathcal{I}_{\mathtt{A}}(P^{\text{key,i}} \mid |\mathcal{G}_{H'}| = 0) = \mathcal{I}_{\mathtt{A}}(P^{\text{key,j}} \mid |\mathcal{G}_{H'}| = 0), \tag{14}$$

where we remind that $|\mathcal{G}_{H'}| = 0$ is the event 0 agents in $s^\star$ at timestep $h = H'$

Let $\Pr(\mathtt{A}(\mathcal{I}_{\mathtt{A}}(P^{\text{key,j}}); r^{\text{key}}) = \widehat{\pi}, \widehat{\pi} = i)$ be the probability the algorithm $\mathtt{A}$ chooses a policy $\widehat{\pi}$ and $\widehat{\pi}$ chooses actions' index i after interaction $\mathcal{I}_{\mathtt{A}}(P^{\text{key,j}})$, where the reward function is r^{key}. We can simply look at the probability of the algorithm to choose sequence i after the interaction: $\Pr(\mathtt{A}(\mathcal{I}_{\mathtt{A}}(P^{\text{key,j}})) = i)$, and we omit the reward for ease of notation.

Note that for any interaction, the algorithm chooses one sequence of the last H' actions (with probability 1), hence,

$$\sum_{i=1}^{A^{H'}} \Pr(\mathtt{A}(\mathcal{I}_{\mathtt{A}}(P^{\text{key},1})) = i \mid |\mathcal{G}_{H'}| = 0) = 1. \tag{15}$$

Since the reward r^{key} is built in such a way that only guessing the full key yields a reward, we get,

$$\begin{aligned}
&\mathbb{E}_{\mathtt{A}}(\mid |\mathcal{G}_{H'}| = 0) \\
&= 1 \cdot \Pr(\mathtt{A}(\mathcal{I}_{\mathtt{A}}(P^{\text{key,i}})) = i \mid |\mathcal{G}_{H'}| = 0) + 0 \cdot \Pr(\mathtt{A}(\mathcal{I}_{\mathtt{A}}(P^{\text{key,i}})) \neq i \mid |\mathcal{G}_{H'}| = 0),
\end{aligned}$$

and we get,

$$\mathbb{E}_{\mathtt{A}}(V^{\widehat{\pi}}(s_0; P^{\text{key,i}}, r^{\text{key}})) \mid |\mathcal{G}_{H'}| = 0) = \Pr(\mathtt{A}(\mathcal{I}_{\mathtt{A}}(P^{\text{key,i}})) = i \mid |\mathcal{G}_{H'}| = 0), \tag{16}$$

where $r_h^{\text{key}}(s, a; P^{\text{key,i}})$ is the reward at dynamics $P^{\text{key,i}}$ at state s timestep h and action a.

Let us define the uniform distribution over the Key-Dynamic with such a fixed first half, and denote this distribution over the environments with $\mathcal{E}$. Denote the expectation over the randomness of the

algorithm with $\mathbb{E}_{\mathtt{A}}$. We get,

$$
\begin{aligned}
&= \mathbb{E}_{P^{\text{key}} \sim \mathcal{E}} \mathbb{E}_{\mathtt{A}} \left(V^{\widehat{\pi}}(s_0; P^{\text{key}}, r^{\text{key}}) \mid |\mathcal{G}_{H'}| = 0 \right) \\
&= \sum_{i=1}^{A^{H'}} \frac{1}{A^{H'}} \mathbb{E}_{\mathtt{A}} \left(V^{\widehat{\pi}}(s_0; P^{\text{key,i}}, r^{\text{key}}) \mid |\mathcal{G}_{H'}| = 0 \right) \\
&= \frac{1}{A^{H'}} \sum_{i=1}^{A^{H'}} \Pr(\mathtt{A}(\mathcal{I}_{\mathtt{A}}(P^{\text{key,i}})) = i \mid |\mathcal{G}_{H'}| = 0) \quad \text{(need to guess the key: Equation (16))} \\
&= \frac{1}{A^{H'}} \sum_{i=1}^{A^{H'}} \Pr(\mathtt{A}(\mathcal{I}_{\mathtt{A}}(P^{\text{key},1})) = i \mid |\mathcal{G}_{H'}| = 0) \quad \text{(same interactions: Equation (14))} \\
&= \frac{1}{A^{H'}} \cdot 1 \quad \text{(one action is being chosen: Equation (15))}
\end{aligned}
$$

Therefore, there exists an dynamics P^{key} which its first actions are $\bar{a}_{H-H'}$ in which the algorithm performs at expectation with at most $\frac{1}{A^{H'}}$ reward, where $r = r^{\text{key}}$. □

Corollary B.10. *With high probability, the algorithm has high error when it does not know the key. Formally, Assume we have only one learning phase, Assume $A \geq 2$. Let $\mathtt{A}$ be an algorithm, for any sequence of $H-1$ actions $\bar{a}_{H-1}$, there exists a Key-Dynamic of horizon H, denoted by P^{key}, with the first $H-1$ good actions as $\bar{a}_{H-1}$, s.t.,*

$$
\mathbb{P}\left(V^{\star}(s_0; P^{\text{key}}, r^{\text{key}}) - V^{\widehat{\pi}}(s_0; P^{\text{key}}, r^{\text{key}}) \geq 0.1 \mid |\mathcal{G}_{H-1}| = 0 \right) \geq \frac{4}{9}.
$$

Proof. From Markov inequality (see Lemma C.2 for the conditional expectation version of Markov inequality) and from Lemma B.9,

$$
\begin{aligned}
&\mathbb{P}\left(V^{\widehat{\pi}}(s_0; P^{\text{key}}, r^{\text{key}}) \geq 0.9 \mid |\mathcal{G}_{H-1}| = 0 \right) \\
&\leq \mathbb{E}\left(V^{\widehat{\pi}}(s_0; r^{\text{key}}, r^{\text{key}}) \mid |\mathcal{G}_{H-1}| = 0 \right) \cdot \frac{10}{9} \\
&\leq \frac{1}{A} \frac{10}{9} \leq \frac{5}{9}.
\end{aligned}
$$

The rest follows from that $V^{\star} = 1$. □

Definition B.11. We define with Z_i^j the event that no agent in phases 0 to $j-1$ reached timestep i in the Key-Dynamic.

Corollary B.12. *Fix algorithm $\mathtt{A}$. there exists a Key-Dynamic with horizon H s.t. for any $i < h \leq H$*

$$
\mathbb{E}\left(|\mathcal{G}_h^j| \mid Z_i^j \right) \leq \frac{m}{A^{h-i}}.
$$

Proof. We reduce the problem to learning with a single phase. Under the event Z_i^j, we can view the part of the dynamics of timesteps after i at phase j as part of the dynamics in the single-phase learning. In this case, the good agents in the single phase problem at timestep i, $\mathcal{G}_i$, are the same as the good agents in the ρ phases problem at phase j at timestep i, $\mathcal{G}_i^j$. The number of agents at $s^{\star}$ at the beginning of phase j is at most m, so $\mathcal{G}_i \leq m$. And the corollary follows from Corollary B.8, when $|\mathcal{G}_i| \leq m$. □

Corollary B.13. *For every algorithm $\mathtt{A}$ with one learning phase, There exists a Key-Dynamic of horizon $H-1$ s.t. for any phase $i \in \{0, \ldots, \rho-1\}$, the probability that there are no agents in the informative state $s^{\star}$ at timestep $i \cdot (H-1)/\rho$ is high, conditioned there were zero agents at phase $i-1$ at $(i-1) \cdot (H-1)/\rho$. Specifically,*

$$
\mathbb{P}\left(Z^i_{(i+1)H'} \mid Z^i_{iH'} \right) > 1 - \frac{m}{A^{H'}},
$$

where $H' := (H-1)/\rho$. We assume for simplicity that ρ divides $H-1$, and a similar result without this assumption can be derived.

Proof. Remember that i starts at 0. Under the event $Z^i_{iH'}$, the event that there are zero agents in timestep $(i+1)H'$ at phase i and the event $Z^i_{(i+1)H'}$, are the same. Hence,

$$\mathbb{P}\left(|\mathcal{G}^i_{(i+1)H'}| = 0 \mid Z^i_{iH'}\right) = \mathbb{P}\left(Z^i_{(i+1)H'} \mid Z^i_{iH'}\right).$$

From Markov inequality (Lemma C.2) and from Corollary B.12,

$$\mathbb{P}\left(|\mathcal{G}^i_{(i+1)H'}| \geq 1 \mid Z^i_{iH'}\right) \leq \frac{m}{A^{H'}}.$$

then,

$$\mathbb{P}\left(|\mathcal{G}^i_{(i+1)H'}| < 1 \mid Z^i_{iH'}\right) = \mathbb{P}\left(|\mathcal{G}^i_{(i+1)H'}| = 0 \mid Z^i_{iH'}\right) > 1 - \frac{m}{A^{H'}}.$$

□

Definition B.14. We denote with Ω the probability space, and we mainly use it to define complementary events. I.e., let G be an event, its complementary event is denoted with $\Omega \setminus G$.

Theorem B.15. *Assume the number of agents satisfies,*

$$m \leq \frac{A^{(H-1)/\rho}}{\rho \cdot 100}.$$

Assume $\rho < H$ and assume for simplicity that ρ divides $H-1$ (otherwise a similar argument with $(A^{\lfloor (H-1)/\rho \rfloor})/\rho$ can be derived). Assume $A \geq 2$. Then, for any algorithm $\mathtt{A}$ with ρ phases of learning, there exists a Key-Dynamic P^{key} with A actions and horizon H s.t.,

$$\mathbb{P}\left(V^\star(s_0; P^{\text{key}}, r^{\text{key}}) - V^{\widehat{\pi}}(s_0; P^{\text{key}}, r^{\text{key}}) \geq 0.1\right) \geq 0.44,$$

where $\widehat{\pi}$ maximizes the value function under reward r^{key} and the learned dynamics.

Proof. We split the analysis into two cases. One is the event in which in every phase the algorithm discovers no more than H' new actions in the key, where $H' := (H-1)/\rho$. The other is the complement event, i.e., there exists a phase in which some agent discovers more than H' new actions in the key. We denote the first event with $\mathcal{Z}$. Note that $\mathcal{Z} = \cap_{i=0}^{\rho-1} Z^i_{(i+1)H'}$.

We built the dynamics from two parts. The first $H-1$ good actions are from the $H-1$ horizon dynamics that Corollary B.13 promises exist. The last good action is from Corollary B.10.

Since we use the first $H-1$ actions as the actions from the Key-Dynamic in Corollary B.13, we get,

$$\mathbb{P}(\Omega \setminus \mathcal{Z}) \leq \rho(\frac{m}{A^{(H-1)/\rho}}) \leq \frac{1}{100},$$

where we used the union bound to bound the probability.

If none of the agents reached state $s^\star$ in timestep $H-1$ in every phase, i.e., $Z^{\rho-1}_{H-1}$ holds, it is the same the single-phase learning where $|\mathcal{G}_{H-1}| = 0$. I.e., we can use Corollary B.10. Hence, for any sequence of $H-1$ actions we have a Key-Dynamic of horizon H, denoted P^{key}, s.t. its prefix key is the sequence of $H-1$ actions, and,

$$\mathbb{P}\left(V^\star(s_0; P^{\text{key}}, r^{\text{key}}) - V^{\widehat{\pi}}(s_0; P^{\text{key}}, r^{\text{key}}) \geq 0.1 \mid |\mathcal{G}_{H-1}| = 0\right) \geq \frac{4}{9}.$$

Notice that for the entire dynamics of horizon H, it still holds that $\mathbb{P}(\mathcal{Z}) \geq \frac{99}{100}$.

$$\begin{aligned}
&\mathbb{P}\left(V^\star(s_0; P^{\text{key}}, r^{\text{key}}) - V^{\widehat{\pi}}(s_0; P^{\text{key}}, r^{\text{key}}) \geq 0.1\right) \\
&= \mathbb{P}\left(V^\star(s_0; P^{\text{key}}, r^{\text{key}}) - V^{\widehat{\pi}}(s_0; P^{\text{key}}, r^{\text{key}}) \geq 0.1 \mid \mathcal{Z}\right)\mathbb{P}(\mathcal{Z}) \\
&+ \mathbb{P}\left(V^\star(s_0; P^{\text{key}}, r^{\text{key}}) - V^{\widehat{\pi}}(s_0; P^{\text{key}}, r^{\text{key}}) \geq 0.1 \mid \Omega \setminus \mathcal{Z}\right)\mathbb{P}(\Omega \setminus \mathcal{Z}) \\
&\geq \mathbb{P}\left(V^\star(s_0; P^{\text{key}}, r^{\text{key}}) - V^{\widehat{\pi}}(s_0; P^{\text{key}}, r^{\text{key}}) \geq 0.1 \mid \mathcal{Z}\right)\mathbb{P}(\mathcal{Z}) + 0 \\
&\geq \frac{4}{9} \cdot \mathbb{P}(\mathcal{Z}) \quad \text{(from Corollary B.10)} \\
&\geq \frac{4}{9} \cdot \frac{99}{100} \geq 0.44.
\end{aligned}$$

□

C Auxiliary Lemmas

Lemma C.1. *Let $P \in \mathbb{R}^{m,n}$ be a matrix s.t. $\forall i$, $\sum_{j=1}^{n} P_{ij} \leq 1$, and each element $P_{ij} \geq 0$. E.g., for some row i an entry P_{ij} is the probability to move from i to j, and for other row i' it is zero for all j. Let $v \in \mathbb{R}^m$. Then,*

$$\|vP\|_1 \leq \|v\|_1.$$

Proof.

$$\begin{aligned}
\|vP\|_1 &= \sum_j |\sum_i v_i P_{ij}| \leq \sum_j \sum_i |v_i P_{ij}| \\
&= \sum_j \sum_i |v_i| P_{ij} \quad \text{(the elements are } \geq 0) \\
&= \sum_i \sum_j |v_i| P_{ij} = \sum_i |v_i| \sum_j P_{ij} \\
&\leq \sum_i |v_i| \cdot 1 \quad \text{(from the assumption)} \\
&= \|v\|_1.
\end{aligned}$$

□

Lemma C.2 (Markov inequality for conditional expectation)**.** *Let $(\Omega, \mathcal{F}, P)$ be a probability space. Let $X \geq 0$ be a non-negative random variable. Let $a, b \in \mathbb{R}, a > 0, b \geq 0$. Let $A \in \mathcal{F}$ be an event.*

Assume $E(X|A) \leq b$.

Then, $\Pr(X \geq a|A) \leq b/a$.

Proof. From the definition of conditional expectation we have,

$$\int_A X dP = \int_A E(X|A) dP,$$

and from the assumption we get,

$$\int_A E(X|A) dP \leq \int_A b dP = bPr(A).$$

Together we get

$$\int_A X dP \leq bPr(A). \tag{17}$$

Hence,

$$Pr(X \geq a|A) = \frac{Pr(X \geq a, A)}{Pr(A)} = \frac{\int_{A \cap \{X \geq a\}} 1 dP}{Pr(A)} \leq \frac{\frac{1}{a}\int_A X dP}{Pr(A)} \leq \frac{\frac{1}{a} bPr(A)}{Pr(A)} = \frac{b}{a},$$

where the second inequality is from Equation (17), and the first inequality is since,

$$\int_{\{X \geq a\} \cap A} 1 dP = \frac{1}{a} \int_{\{X \geq a\} \cap A} a dP \leq \frac{1}{a} \int_{\{X \geq a\} \cap A} X dP \leq \frac{1}{a} \int_A X dP.$$

□

Lemma C.3 (Bretagnolle–Huber–Carol, Lemma 10.13 in [22])**.** *Let X be a random variable taking values in $\{1, \ldots, S\}$, where $\Pr[X = i] = p_i$. Assume we sample X for n times and observe the value i in $\hat{n}_i$ outcomes. Then,*

$$\Pr\left[\sum_{i=1}^{k} \left| \frac{\hat{n}_i}{n} - p_i \right| \geq \lambda \right] \leq 2^{k+1} e^{-2n\lambda^2}.$$

Corollary C.4 (Bretagnolle–Huber–Carol with a $2S$ term)**.** *Let X be a random variable taking values in $\{1, \ldots, S\}$, where $\Pr[X = i] = p_i$. Assume we sample X for n times and observe the value i in $\hat{n}_i$ outcomes. Denote the empirical distribution with $\hat{p}$, i.e., $\hat{p}_i = \hat{n}_i/n$ Then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$,*

$$\|\hat{p} - p\|_1 \;\leq\; \sqrt{\frac{\ln(1/\delta) + 2S}{2n}}.$$

Proof. The Bretagnolle–Huber–Carol inequality gives, for all $\lambda > 0$,

$$\Pr(\|\hat{p} - p\|_1 \geq \lambda) \;\leq\; 2^{S+1} e^{-2n\lambda^2}.$$

Set the right-hand side to be at most δ:

$$2^{S+1} e^{-2n\lambda^2} \leq \delta \iff 2n\lambda^2 \geq \ln(1/\delta) + (S+1)\ln 2.$$

Since $S \geq 1$, we have $(S+1)\ln 2 \leq 2S$, hence it suffices that,

$$2n\lambda^2 \geq \ln(1/\delta) + 2S,$$

which yields the stated bound. □

Lemma C.5 (Bretagnolle–Huber–Carol with a random variable)**.** *Let $\mathcal{K}$ be a random variable with support of cardinality $\mathrm{supp}(\mathcal{K})$. Let X be a random variable taking values in $\{1, \ldots, S\}$, where $\Pr[X = i] = p_i$. Assume we sample X for $\mathcal{K}$ times and observe the value i in $\hat{\mathcal{K}}_i$ outcomes. Denote the empirical distribution with $\hat{p}$, i.e., $\hat{p}_i = \hat{\mathcal{K}}_i/\mathcal{K}$ Then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$,*

$$\|\hat{p} - p\|_1 \;\leq\; \sqrt{\frac{\log(\mathrm{supp}(\mathcal{K})/\delta) + 2S}{2\mathcal{K}}}.$$

Proof. Define the threshold $\lambda_{\mathcal{K}}$ as:

$$\lambda_{\mathcal{K}} = \sqrt{\frac{\log(\mathrm{supp}(\mathcal{K})/\delta) + 2S}{2\mathcal{K}}}.$$

We evaluate the probability by conditioning on the random variable $\mathcal{K}$:

$$\begin{aligned}\Pr\left(\|p - \hat{p}\|_1 \geq \lambda_{\mathcal{K}}\right) &= \sum_k \Pr\left(\|p - \hat{p}\|_1 \geq \lambda_{\mathcal{K}} \mid \mathcal{K} = k\right) \Pr(\mathcal{K} = k) \\ &\leq \sum_k \left(2^{S+1} e^{-2k\lambda_k^2}\right) \Pr(\mathcal{K} = k),\end{aligned}$$

where the last inequality holds since for a fixed $\mathcal{K} = k$, we have a bound which is not a random variable, but rather a number, hence, we can use Lemma C.3.

We take the logarithm:

$$\begin{aligned}&\log \mathrm{supp}(\mathcal{K}) + (S+1)\log 2 - \sum_k 2k\lambda_k^2 \Pr(\mathcal{K} = k) \\ &= \log \mathrm{supp}(\mathcal{K}) + (S+1)\log 2 - \sum_k 2k \left(\frac{\log(\mathrm{supp}(\mathcal{K})/\delta) + 2S}{2k}\right) \Pr(\mathcal{K} = k). \\ &= \log \mathrm{supp}(\mathcal{K}) + (S+1)\log 2 - (\log(\mathrm{supp}(\mathcal{K})/\delta) + 2S) \quad (\text{since } \sum_k \Pr(\mathcal{K} = k) = 1) \\ &\leq \log \mathrm{supp}(\mathcal{K}) + 2S - \log \mathrm{supp}(\mathcal{K}) + \log \delta - 2S \\ &= \log \delta.\end{aligned}$$

Hence,

$$\Pr(\|p - \hat{p}\|_1 \geq \lambda_{\mathcal{K}}) \leq \delta.$$

□

Lemma C.6 (Chernoff–Hoeffding, Lemma 10.2.1 in [22])**.** *Let $R_1, \ldots, R_m$ be m i.i.d. samples of a random variable $R \in [0,1]$. Let $\mu = \mathbb{E}[R]$ and $\hat{\mu} = \frac{1}{m}\sum_{i=1}^{m} R_i$. For any $\varepsilon \in (0,1)$, we have*

$$\Pr\left[|\mu - \hat{\mu}| \geq \varepsilon\right] \leq 2e^{-2\varepsilon^2 m}.$$

In addition,

$$\Pr[\hat{\mu} \leq (1-\varepsilon)\mu] \leq e^{-\varepsilon^2 \mu m/2} \qquad \textit{and} \qquad \Pr[\hat{\mu} \geq (1+\varepsilon)\mu] \leq e^{-\varepsilon^2 \mu m/3}.$$

We will refer to the first bound as additive *and the second set of bounds as* multiplicative.

Lemma C.7 (Based on Theorem 10.9 in [22])**.** *Let q_1 and q_2 be two distributions over $\mathcal{S}$. Let $f : \mathcal{S} \cup \{s_{\text{sink}}\} \to [0, F_{\max}]$ and $f(s_{\text{sink}}) = 0$. Then,*

$$|\mathbb{E}_{s\sim q_1}[f(s)] - \mathbb{E}_{s\sim q_2}[f(s)]| \leq F_{\max} \|q_1 - q_2\|_1,$$

where the norm is just over $\mathcal{S}$:

$$\|q_1 - q_2\|_1 = \sum_{s\in\mathcal{S}} |q_1(s) - q_2(s)|.$$

Proof. Based on [22],

$$\begin{aligned}
|\mathbb{E}_{s\sim q_1}[f(s)] - \mathbb{E}_{s\sim q_2}[f(s)]| &= \left| \sum_{s\in\mathcal{S}\cup\{s_{\text{sink}}\}} f(s)q_1(s) - \sum_{s\in\mathcal{S}} f(s)q_2(s) \right| \\
&= \left| \sum_{s\in\mathcal{S}} f(s)q_1(s) - \sum_{s\in\mathcal{S}} f(s)q_2(s) \right| \quad (\text{since } f(s_{\text{sink}}) = 0) \\
&\leq \sum_{s\in\mathcal{S}} f(s)\,|q_1(s) - q_2(s)| \\
&\leq F_{\max}\,\|q_1 - q_2\|_1.
\end{aligned}$$

□

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction accurately reflect the paper's contributions and scope.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discuss the limitations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: The paper provides the full set of assumptions and complete (and correct) proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [N/A]

   Justification: The paper does not include experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [N/A]

Justification: The paper does not include experiments.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [N/A]

   Justification: The paper does not include experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [N/A]

   Justification: The paper does not include experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [N/A]

   Justification: The paper does not include experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: The paper does not have societal impacts beyond those of general ML research.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [N/A]

    Justification: The paper does not use existing assets.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: the paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.