Tempo Adaption in Non-stationary Reinforcement Learning

Hyunin Lee[1] **Yuhao Ding**[1] **Jongmin Lee**[1]
Ming Jin[2] **Javad Lavaei**[1] **Somayeh Sojoudi**[1]
[1]UC Berkeley [2]Virginia Tech
{hyunin,yuhao_ding,jongmin.lee,lavaei,sojoudi}@berkeley.edu
jinming@vt.edu

Abstract

We first raise and tackle "time synchronization" issue between the agent and the environment in non-stationary reinforcement learning (RL), a crucial factor hindering its real-world applications. In reality, environmental changes occur over wall-clock time (t) rather than episode progress (k), where wall-clock time signifies the actual elapsed time within the fixed duration $\mathrm{t} \in [0, T]$. In existing works, at episode k, the agent rollouts a trajectory and trains a policy before transitioning to episode $k+1$. In the context of the time-desynchronized environment, however, the agent at time t_k allocates $\Delta\mathrm{t}$ for trajectory generation and training, subsequently moves to the next episode at $\mathrm{t}_{k+1} = \mathrm{t}_k + \Delta\mathrm{t}$. Despite a fixed total episode (K), the agent accumulates different trajectories influenced by the choice of *interaction times* ($\mathrm{t}_1, \mathrm{t}_2, ..., \mathrm{t}_K$), significantly impacting the sub-optimality gap of policy. We propose a Proactively Synchronizing Tempo (`ProST`) framework that computes optimal $\{\mathrm{t}_1, \mathrm{t}_2, ..., \mathrm{t}_K\} (= \{\mathrm{t}\}_{1:K})$. Our main contribution is that we show optimal $\{\mathrm{t}\}_{1:K}$ trades-off between the policy training time (agent tempo) and how fast the environment changes (environment tempo). Theoretically, this work establishes an optimal $\{\mathrm{t}\}_{1:K}$ as a function of the degree of the environment's non-stationarity while also achieving a sublinear dynamic regret. Our experimental evaluation on various high dimensional non-stationary environments shows that the `ProST` framework achieves a higher online return at optimal $\{\mathrm{t}\}_{1:K}$ than the existing methods.

1 Introduction

The prevailing reinforcement learning paradigm gathers past data, trains models in the present, and deploys them in the *future*. This approach has proven successful for *stationary* Markov decision processes (MDPs), where the reward and transition functions remain constant [1–3]. However, challenges arise when the environments undergo significant changes, particularly when the reward and transition functions are dependent on time or latent factors [4–6], in *non-stationary* MDPs. Managing non-stationarity in environments is crucial for real-world reinforcement learning (RL) applications. Thus, adapting to changing environments is pivotal in non-stationary RL.

This paper addresses a practical concern that has inadvertently been overlooked within traditional non-stationary RL environments, namely, the time synchronization between the agent and the environment. We raise the impracticality of utilizing *episode-varying* environments in existing non-stationary RL research, as such environments do not align with the real-world scenario where changes occur regardless of the agent's behavior. In an episode-varying environment, the agent has complete control over determining the time of executing episode k, the duration of policy training between the episode $k, k+1$, and the time of transition to the episode $k+1$. The issue stems from the premise that the

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

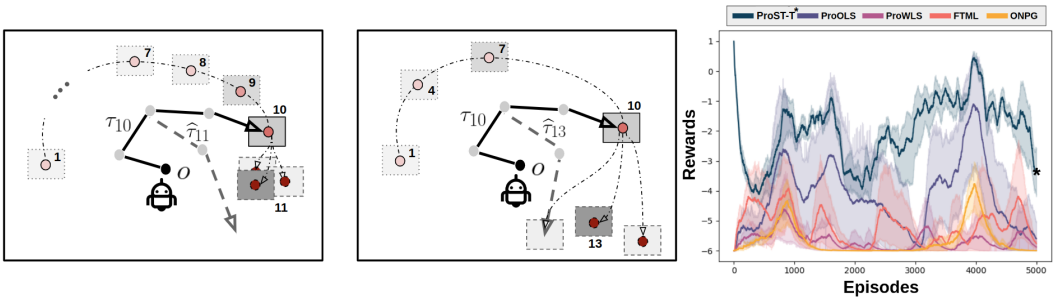

Figure 1: (a) 2D goal reacher in a time-desynchronized environment for one policy update. The agent learns an inaccurate policy on an accurate model (b) For three policy updates, the agent learns an near-optimal policy on an inaccurate model. (c) Rewards per episode in 2D goal reacher with four model-free baselines, where ProST-T* is one of our proposed methods.

environment undergoes dynamic changes throughout the course of each episode, with the rate of non-stationarity contingent upon the behavior exhibited by the agent. However, an independent *wall-clock time* (t) exists in real world environment, thereby above three events are now recognized as wall-clock time t_k, training time $\Delta\mathrm{t}$, and t_{k+1}. The selection of interaction times $(\mathrm{t}_k, \mathrm{t}_{k+1})$ has a notable impact on the trajectories collected, while the interval $\mathrm{t}_{k+1} - \mathrm{t}_k$ establishes an upper limit on the duration of training ($\Delta\mathrm{t}$). This interval profoundly influences the sub-optimality gap of the policy. In the context of the time-desynchronized environment, achieving an optimal policy necessitates addressing a previously unexplored question: the determination of the *optimal time sequence* $\{\mathrm{t}_1, \mathrm{t}_2, .., .\mathrm{t}_K\} (= \{\mathrm{t}\}_{1:K})$ when the agent interacts with the environment.

We elucidate the significance of the aforementioned research question through the following example. Consider a robot with the goal of reaching inside a gray-shaded nonfixed target box, known as the goal reacher (Appendix A.1). Note that the reward changes as the box's position shifts over time (Figure 1-(a)). We begin by considering a scenario in which wall-clock time and episode are synchronized, wherein the environment evolves alongside the episode. During each episode k, the agent rollouts a trajectory and iteratively updates the policy N times, with the assumption that one policy update requires one second, then the agent transitions to the subsequent episode $k+1$. In conventional non-stationary RL environments, it is evident that a larger value of N provides an advantage in terms of faster adaptation for achieving a near-optimal policy. However, regardless of the chosen value of N, the agent will consistently encounter the same environment in the subsequent episode. Now, consider a scenario in which wall-clock time and episode are desynchronized. In this context, given a fixed wall-clock time duration $\mathrm{t} \in [0, 10]$, the agent is faced with the additional tasks of determining both the total number of interactions (denoted as total episode K) and the specific time for these interactions $\{\mathrm{t}\}_{1:K}$ where $\mathrm{t}_k \in [0, 10], \mathrm{t}_{k-1} < \mathrm{t}_k$ for $\forall k \in [K]$. Figure 1-(a) shows an agent that interacts with the environment ten times, i.e. $\mathrm{t} = [1, 2, ..., 10]$, and spends interval (t_k, t_{k+1}) to train the policy, which consumes a second $(K = 10, N = 1)$. High interaction frequency $(K = 10)$ provides adequate data for precise future box position learning $(\mathrm{t} = 11)$, yet a single policy update $(N = 1)$ might not approximate optimal policy. Now, if the agent interacts with the environment four times, i.e. $\mathrm{t} = [1, 4, 7, 10]$ (see Figure 1-(b)), it becomes feasible to train the policy over a duration of three seconds $(K = 4, N = 3)$. A lengthier policy training period $(N = 3)$ aids the agent in obtaining near-optimal policy. However, limited observation data $(K = 4)$ and significant time intervals $(\mathrm{t} = 11, 12, 13)$ might lead to less accurate box predictions. This example underscores the practical importance of aligning the agent's interaction timing with the environment in non-stationary RL. Determining the optimal $\mathrm{t}_{1:K}$ involves a trade-off between achieving an optimal model and an optimal policy.

Based on the previous example, our key insight is that, in non-stationary environments, a new factor, **tempo** emerges. Informally, tempo refers to the pace of processes occurring in a non-stationary environment. We define **environment tempo** as how fast the environment changes, and **agent tempo** as how frequently it updates the policy. Despite the importance of considering the tempo to find the optimal $\{\mathrm{t}\}_{1:K}$, the existing formulations and methods for non-stationarity RL are insufficient. None of the existing works have adequately addressed this crucial aspect.

Our framework, ProST, provides a solution to find optimal $\{\mathfrak{t}\}_{1:K}$. It proactively optimizes time sequence by leveraging the agent tempo and the environment tempo. ProST framework is divided into two components: Future policy optimizer ($\texttt{OPT}_\pi$) and time optimizer ($\texttt{OPT}_\mathfrak{t}$), and is characterized by three key features: 1) it is *proactive* in nature as it forecasts the future MDP model; 2) it is *model-based* as it optimizes the policy in the created MDP; and 3) it is *synchronizing tempo* framework as it finds optimal training time by adjusting how many times the agent needs to update the policy (agent tempo) to how fast the environment changes (environment tempo). Our framework is general in the sense that it allows to be equipped with any algorithms to update the policy. Compared to the existing works [7–9], our approach achieves both higher rewards and more stable performance over time (Figure 1-(c) and Section 5).

We analyze the statistical and computational properties of ProST in tabular MDP, which named as ProST-T. Our framework learns in a novel MDP, elapsed time-varying MDP, and quantifies its non-stationarity with a novel metric, time-elapsing variation budget, where both consider an actual time taken. We develop the dynamic regret of ProST-T (Theorem 1) into two components: dynamic regret of $\texttt{OPT}_\pi$ that learns a future MDP model (Proposition 1) and that of $\texttt{OPT}_\mathfrak{t}$ that computes a near-optimal policy in that model (Theorem 2, Proposition 2). We show that both regrets can satisfy a sublinear to the total episode regardless of agent tempo. Perhaps most importantly, we show that an optimal training time is a minimizer of trade-off problem between those two dynamic regrets, with each reflecting the tempo of the agent and the environment (Theorem 3). We find an interesting property that future MDP model error of $\texttt{OPT}_\pi$ serves as a common factor on both regrets and shows tight dynamic regret of ProST-T can be achieved with joint optimization problem of learning both different weights on observed data and a model. (Theorem 4, Remark 1).

Finally, we introduce ProST-G, an adaptable learning algorithm for high-dimensional tasks achieved through a practical approximation of ProST. Empirically, ProST-G provides solid evidence on different reward returns depending on policy training time and the significance of learning the future MDP model. ProST-G also consistently finds a near-optimal policy, outperforming four popular RL baselines that are used in non-stationary environments on three different Mujoco tasks.

Notations

The sets of natural, real, and nonnegative real numbers are denoted by $\mathbb{N}, \mathbb{R}, \mathbb{R}_+$ respectively. We denote For a finite set Z, the notation $|Z|$ denotes its cardinality, and the notation $\Delta(Z)$ denotes the probability simplex over Z. For $X \in \mathbb{N}$, we define $[X] = \{1, 2, .., X\}$. For any variable X, we denote $\widehat{X}$ as a *forecasted*(or *predicted*) variables at current time, $\widetilde{X}$ as observed value at past. Also, for any time variable $\mathfrak{t} > 0$ and $k \in \mathbb{N}$, we denote time sequence $\{\mathfrak{t}_1, \mathfrak{t}_2, .., \mathfrak{t}_k\}$ as $\{\mathfrak{t}\}_{1:k}$, and variable X at time $\mathfrak{t}_k$ as $X_{\mathfrak{t}_k}$. We use short notation $X_{(k)}$ for $X_{\mathfrak{t}_k}$. We denote $\{x\}_{a:b}$ as a sequence of variables $\{x_a, x_{a+1}, ..., x_b\}$, and $\{x\}_{(a:b)}$ as a sequence of variables $\{x_{\mathfrak{t}_a}, x_{\mathfrak{t}_{a+1}}, ..., x_{\mathfrak{t}_b}\}$. Also, for any variables x, y, we denote $x \vee y$ as $\max(x, y)$, and $x \wedge y$ as $\min(x, y)$. Last, for any complex numbers z_1, z_2, we denote $z_2 = W(z_1)$ as Lambert W function that holds if only if $z_2 e^{z_2} = z_1$ holds. We have described the specific details in Appendix C.1.

2 Problem statement: desynchronising timelines

2.1 Time elapsing Markov Decision Process

In this paper, we study a non-stationary Markov Decision Process (MDP) that transition probability and reward change. We begin by clarifying that the term *episode* is agent-centric, not environment-centric. Prior solutions for episode-varying (or step-varying) MDPs operate under the assumption that the timing of MDP changes aligns with the agent commencing a new episode (or step).

Therefore, we introduce a novel concept **time elapsing MDP**. It starts from wall-clock time $\mathfrak{t} = 0$ to $\mathfrak{t} = T$ where T is fixed. Time elapsing MDP at time $\mathfrak{t} \in [0, T]$ is defined as $\mathcal{M}_\mathfrak{t} := \langle \mathcal{S}, \mathcal{A}, H, P_\mathfrak{t}, R_\mathfrak{t}, \gamma \rangle$ where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, H is the number of steps, $P_\mathfrak{t} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \Delta(\mathcal{S})$ is the transition probability , $R_\mathfrak{t} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, and γ is a discounting factor. Prior to executing the first episode, the agent determines the total number of interactions with the environment (denoted as total episode K) and subsequently computes the sequence of interaction times $\{\mathfrak{t}\}_{1:K}$ through an optimization problem. We denote $\mathfrak{t}_k$ as the wall-clock time of the environment when the agent starts an episode k. Now, same as existing non-stationary RL problem framework, the agent's objective is learning a policy $\pi^{\mathfrak{t}_k} : \mathcal{S} \to \Delta(\mathcal{A})$ for all k. This is achieved through engaging in a total of

K episodes involving interactions with $\{\mathcal{M}_{\mathfrak{t}_1}, \mathcal{M}_{\mathfrak{t}_1}, ..., \mathcal{M}_{\mathfrak{t}_K}\}$ where the agent dedicates time interval $(\mathfrak{t}_k, \mathfrak{t}_{k+1})$ for policy training, then finally obtain a sequence of optimal policies $\{\pi_{\mathfrak{t}_1}, \pi_{\mathfrak{t}_2}, ..., \pi_{\mathfrak{t}_K}\}$ to maximize non-stationary policy evaluation metric, *dynamic regret*.

It is straightforward that the replacing time elapsing MDP with conventional MDP raises a significant question that should be addressed before the agent's initial execution: within a time duration $[0, T]$, which time sequence $\{\mathfrak{t}\}_{1:K}$ yields favorable trajectory samples to obtain optimal policy, but perhaps most crucially, what is optimal interaction episode K^* that encompasses the delicate balance between amount of observed trajectories and accuracy of policy training. This question stands apart from prevailing RL objective, computation of optimal policy for given $\mathfrak{t}_k$. In this paper, we propose `ProST` framework that computes optimal K^* and corresponding optimal time sequence $\{\mathfrak{t}^*\}_{1:K^*}$ based on the information of the environment's rate of change in Section 4, then computes near-optimal policy for $\{\mathfrak{t}^*\}_{1:K^*}$ in Section 3. In advance of these discussions, we introduce a new metric quantifying the environment's pace of change, time elapsing variation budget.

2.2 Time elapsing variation budget

Variation budget [10–12] is a metric to quantify how fast the environment changes. Driven by our motivations, we introduce a fresh metric imbued with real-time considerations, *time elapsing variation budget* $B(\Delta\mathfrak{t})$. Unlike the existing variation budget which quantifies the environment's non-stationary across $\{1, 2, .., K\}$ episodes, our definition accesses it it across $\{\mathfrak{t}_1, \mathfrak{t}_2, ..., \mathfrak{t}_K\}$, where interval $\Delta\mathfrak{t} = \mathfrak{t}_{k+1} - \mathfrak{t}_k$ remains constant regardless of $k \in [K-1]$. For further analysis, we define two time elapsing variation budgets, one for transition probability and the other for reward function.

Definition 1 (Time elapsing variation budget)**.** *For given $\{\mathfrak{t}_1, \mathfrak{t}_2, .., \mathfrak{t}_K\}$, assume interval $\Delta\mathfrak{t}$ is equal to the policy training time Δ_π. We define two time elapsing variation budgets $B_p(\Delta_\pi), B_r(\Delta_\pi)$ as*

$$B_p(\Delta_\pi) \coloneqq \sum_{k=1}^{K-1} \sup_{s,a} \|P_{\mathfrak{t}_{k+1}}(\cdot\,|s,a) - P_{\mathfrak{t}_k}(\cdot\,|s,a)\|_1,\ B_r(\Delta_\pi) \coloneqq \sum_{k=1}^{K-1} \sup_{s,a} |R_{\mathfrak{t}_{k+1}}(s,a) - R_{\mathfrak{t}_k}(s,a)|.$$

To enhance the representation of a real-world system by the time elapsing variation budget, we introduce the following assumption.

Assumption 1 (Drifting constants)**.** *There exists constants $c, \alpha_p, \alpha_r > 0$ that satisfies $B_p(c\Delta_\pi) = c^{\alpha_p} B_p(\Delta_\pi)$ and $B_r(c\Delta_\pi) = c^{\alpha_r} B_r(\Delta_\pi)$. We call α_p, α_r as drifting constants*

Assumption 1 expands the overage of the time elapsing variation budget to encompass a wider array of scenarios commonly used in practice. For instance, a stationary environment satisfies $\alpha_p = \alpha_r = 0$, and a linear drifting environment satisfies $\alpha_r = \alpha_p = 1$. We employ the time elapsing variation budget to represent the *environment tempo*, denoting it as $B(\Delta_\pi) = (B_r(\Delta_\pi), B_p(\Delta_\pi))$

2.3 Optimal training time

Aside from formal MDP framework, the agent can be informed of varying time elapsing variation budget based on training time $\Delta_\pi \in (0, T)$ even within the same time-elapsing MDP. Intuitively, short Δ_π is inadequate to obtain near-optimal policy, yet it facilitates frequent interactions with the environment, leading to a reduction in empirical model error owing to the larger volume of data. Conversely, longer Δ_π may ensure to obtain near-optimal policy but also introduce greater uncertainty in comprehending the environment. This inspection prompts the following conjecture: presence of an **optimal training time** $\Delta_\pi^* \in (0, T)$ that strikes a balance between the sub-optimal gap of the policy and the empirical model error. If it exists, then Δ_π^* provides an optimal $K^* = \lfloor T/\Delta_\pi^* \rfloor$, and an optimal time sequence where $\mathfrak{t}_k^* = \mathfrak{t}_1 + \Delta_\pi^* \cdot (k-1)$ for $k \in [K^*]$. Our `ProST` framework computes an Δ_π^* and sets $\{\mathfrak{t}^*\}_{1:K^*}$, then finds a *future* near-optimal policy of time $\mathfrak{t}_{k+1}^*$ at time $\mathfrak{t}_k^*$. We first introduce how to find one episode ahead optimal policy $\pi^{*,\mathfrak{t}_{k+1}}$ at time $\mathfrak{t}_k$ when $\{\mathfrak{t}\}_{1:K}$ is given in the following section.

3 Future policy optimizer

For given $\mathfrak{t}_k, \mathfrak{t}_{k+1}$, future policy optimizer ($\texttt{OPT}_\pi$), one module of `ProST` framework, computes a optimal policy of $\mathfrak{t}_{k+1}$ at $\mathfrak{t}_k$ into two procedures: first, forecasts the future MDP model of time $\mathfrak{t}_{k+1}$ at

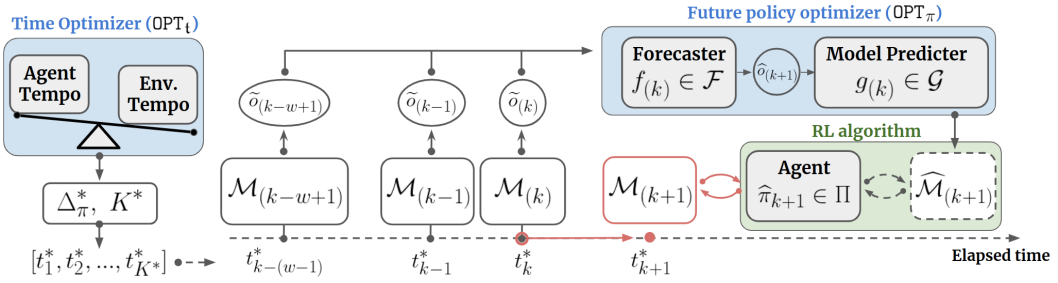

Figure 2: ProST framework

time t_k utilizing MDP forecaster function; second, utilize any policy optimization algorithms within the forecasted MDP model. $\texttt{OPT}_\pi$ to obtain future near-topmnial policy $\pi^{*,t_{k+1}}$.

3.1 MDP forecaster

Our ProST framework is applicable in a environment that meet the following assumption.

Assumption 2 (Observable non-stationary set $\mathcal{O}$)**.** *Let the non-stationarity of $\mathcal{M}_{t_k}$ be fully characterized by a non-stationary variable $o_{t_k} \in \mathcal{O}$. Then, at the end of episode $k \in [K]$ (at time t_k), the agent observes a noisy non-stationary variable $\tilde{o}_{t_k}$.*

It is worth noting that Assumption 2 is mild, as prior research in non-stationary RL has proposed techniques to estimate $o_{(k)}$ through latent factor identification methods [4, 13–16], and our framework accommodates the incorporation of their work for $o_{(k)}$ estimation task. Based on Assumption 2, we define the MDP forecaster function $g \circ f$.

Definition 2 (MDP forecaster $g \circ f$)**.** *Let the function classes $\mathcal{F}$ and $\mathcal{G}$ satisfy $\mathcal{F} : \mathcal{O}^w \to \mathcal{O}$ where $w \in \mathbb{N}$, $\mathcal{G} : \mathcal{S} \times \mathcal{A} \times \mathcal{O} \to \mathbb{R} \times \Delta(\mathcal{S})$. Then, for $f_{(k)} \in \mathcal{F}$ and $g_{(k)} \in G$, we define MDP forecaster at time t_k as $(g \circ f)_{(k)} : \mathcal{O}^w \times \mathcal{S} \times \mathcal{A} \to \mathbb{R} \times \Delta(\mathcal{S})$*

The function $f_{(k)}$, a non-stationarity forecaster, predicts a non-stationary variable $\hat{o}_{(k+1)}$ at time t_{k+1} based on the w past observed variables set $\{\tilde{o}\}_{(k-w+1:k)}$, i.e. $\hat{o}_{(k+1)} = f(\{\tilde{o}\}_{(k-w+1,k)})$. The agent can determine the length of the past reference, denoted as w, by leveraging information from the environment. (Section 4). Then, the function $g_{(k)}$, a model predicter, takes any state and action pair (s, a) with $\hat{o}_{(k+1)}$ and predicts a reward $\widehat{R}_{(k+1)}(s, a)$ and a transition probability $\widehat{P}_{(k+1)}(\cdot|s, a)$ of time t_{k+1}, i.e. $(\widehat{R}_{(k+1)}, \widehat{P}_{(k+1)}) = g_{(k)}(s, a, \hat{o}_{k+1})$. Finally, the $\texttt{OPT}_\pi$ generates the estimated future MDP $\widehat{\mathcal{M}}_{(k+1)} = \langle \mathcal{S}, \mathcal{A}, H, \widehat{P}_{(k+1)}, \widehat{R}_{(k+1)}, \gamma \rangle$ at time t_{k+1}.

3.2 Find future optimal policy

Now, consider an arbitrary RL algorithm which will be used to obtain optimal policy from the model $\widehat{\mathcal{M}}_{(k+1)}$. For given time sequence $\{t\}_{1:K}$, the $\texttt{OPT}_\pi$ finds an future optimal policy as follows: (1) observe and forecast, (2) optimize in future MDP model.

(1) Observe and forecast. At time t_k, the agent executes an episode k in the environment $\mathcal{M}_{(k)}$, completes its trajectory $\tau_{(k)}$, and observes the noisy non-stationary variable $\hat{o}_{(k)}$ (Assumption 2). Then, the algorithm updates the function $f_{(k)}$ based on w past observed variables, and the function $g_{(k)}$ with input from all previous trajectories. Following these updates, the MDP forecaster at time t_k predicts $\widehat{P}_{(k+1)}, \widehat{R}_{(k+1)}$, thereby creating the MDP model for time t_{k+1}, $\widehat{\mathcal{M}}_{(k+1)}$.

(2) Optimize in future MDP model. Up until time t_{k+1}, the agent continually update the policy within the estimated future MDP $\widehat{\mathcal{M}}_{(k+1)}$ for given duration Δ_π. Specifically, the agent rollouts synthetic trajectories $\hat{\tau}_{(k+1)}$ in $\widehat{\mathcal{M}}_{(k+1)}$, then utilizes any policy update algorithm to obtain a policy $\widehat{\pi}_{(k+1)}$. Following the duration Δ_π and time t_{k+1} is reached, the agent stops training and moves to the next episode $\mathcal{M}_{(k+1)}$ with policy $\widehat{\pi}_{(k+1)}$.

We elaborate on the procedure in Algorithm 1 in Appendix F.1.

4 Time optimizer

4.1 Theoretical analysis

We now present our main theoretical contribution, time optimizer ($\texttt{OPT}_\mathfrak{t}$): computing optimal policy training time Δ_π^*(the agent tempo). Our theoretical analysis starts from specifying $\texttt{OPT}_\pi$ optimizer's components, which we refer to as ProST-T (-T stands for an instance in tabular setting). We employ Natural Policy Gradient (NPG) with entropy regularization [17] as a policy update algorithm of OPT_π. We denote entropy regularization coefficient as τ, the learning rate as η, and the policy evaluation approximation gap as δ arising due to finite samples. Without loss of generality, we assume one policy iteration takes a second, i.e. $\Delta_\pi = G$. The theoretical analysis is conducted within a tabular environment, allowing us to relax Assumption 2, which means we estimate non-stationary variables by visitation count of state and action pair $n_{(k)}(s,a)$, rather than observe them. Additionally, we incorporate the exploration bonus term $\Gamma_{(k)}(s,a)$ on $\widehat{R}_{(k+1)}$ to promote the exploration of less frequently visited states and actions.

Equipped with the specific components of ProST-T, we show that the optimal Δ_π^* is the minimizer of the ProST-T's dynamic regret. The dynamic regret of ProST-T is characterized by what we called, *model prediction error*, which measures the MDP forecaster's error by defining the difference between $\widehat{\mathcal{M}}_{(k+1)}$ and $\mathcal{M}_{(k+1)}$ through a Bellman equation.

Definition 3 (Model prediction error)**.** *At time $\mathfrak{t}_k$, MDP forecaster generates the estimated model $\widehat{\mathcal{M}}_{(k+1)}$ and let $\widehat{\pi}^{(k+1)}$ is the optimal policy that obtained from $\widehat{\mathcal{M}}_{(k+1)}$. For any (s,a), denote the state value function and state action value function of $\widehat{\pi}^{[k+1]}$ in $\widehat{\mathcal{M}}_{(k+1)}$ at step $h \in [H]$ as $\widehat{V}_h^{(k+1)}(s)$ and $\widehat{Q}_h^{(k+1)}(s,a)$. Then, we define model prediction error $\iota_h^{k+1}(s,a)$ as follows.*

$$\iota_h^{k+1}(s,a) = \left(R_{(k+1)} + \gamma P_{(k+1)}\widehat{V}_{h+1}^{(k+1)} - \widehat{Q}_h^{(k+1)}\right)(s,a)$$

We now introduce the dynamic regret of ProST-T. At first glance, the dynamic regret of ProST-T is likely controlled by the accuracy of the MDP forecaster's predictions of future MDP model and the capability of the NPG algorithm to approximate the optimal policy within an estimated future MDP model.

Theorem 1 (ProST-T dynamic regret $\mathfrak{R}$)**.** *Let $\iota_H^K = \sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\iota_h^{k+1}(s_h^{k+1}, a_h^{k+1})$ and $\bar{\iota}_\infty^K \coloneqq \sum_{k=1}^{K-1}\|\bar{\iota}_\infty^{k+1}\|_\infty$, where ι_H^K is a data-dependent error. For given $p \in (0,1)$, the dynamic regret of the forecasted policies $\{\widehat{\pi}^{k+1}\}_{1:K-1}$ of ProST-T is upper bounded with probability $1-p/2$,*

$$\mathfrak{R}\left(\{\widehat{\pi}^{k+1}\}_{1:K-1}, K)\right) \le \mathfrak{R}_I + \mathfrak{R}_{II} + C_I[p]\cdot\sqrt{K-1}$$

where $\mathfrak{R}_I = \bar{\iota}_\infty^K/(1-\gamma) - \iota_H^K$, $\mathfrak{R}_{II} = C_{II}[\Delta_\pi]\cdot(K-1)$, and $C_I[p], C_{II}[\Delta_\pi]$ are functions of p, Δ_π, respectively.

This insight is clearly articulated in the Theorem 1. Specifically, the ProST-T dynamic regret is composed of two terms: $\mathfrak{R}_I$ which originates from MDP forecaster error between $\mathcal{M}_{(k+1)}$ and $\widehat{\mathcal{M}}_{(k+1)}$, and $\mathfrak{R}_{II}$ which arises due to the sub-optimality gap between $\pi^*_{(k+1)}$ and $\widehat{\pi}_{(k+1)}$. Theorem 1 clearly demonstrates that prudent construction of MDP forecaster that controls model prediction errors and the selection of the agent tempo Δ_π are significant to guarantee the sublinear $\mathfrak{R}_I, \mathfrak{R}_{II}$, respectively. To incorporate the environment tempo to the $\mathfrak{R}_I$, we observe that the MDP forecaster utilizes past w observations which inherently encapsulates the environment tempo. Then we expect model prediction errors, at least in part, to be controlled by the environment tempo $B(\Delta_\pi)$, so the trade-off between two tempos can be framed as the trade-off between $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$, and finally anticipate there exists optimal Δ_π^* that strikes balance between $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$.

4.1.1 $\mathfrak{R}_{II}$ analysis

The naive way to obtain optimal Δ_π^* is the taking minimum of $\mathfrak{R}_I + \mathfrak{R}_{II}$ and obtain its numerical minimizer by optimization. However, as we mentioned above, it is significant that optimal Δ_π^* should be in a subset of $\mathbb{N}$ that guarantees sublinearity of both $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$ to total episode K. We first compute the set $\mathbb{N}_{II} \subset \mathbb{N}$ that partially bounds Δ_π to guarantee sublinear $\mathfrak{R}_{II}$ then we compute the set $\mathbb{N}_I \subset \mathbb{N}$ that guarantees sublinear $\mathfrak{R}_I$, and finally compute optimal Δ_π^* in a common set $\mathbb{N}_I \cap \mathbb{N}_{II}$.

Proposition 1 (Δ_π bounds for sublinear $\mathfrak{R}_{II}$)**.** *From the given MDP, we have a fixed horizon H. For any $\epsilon > 0$ that satisfies $H = \Omega\left(\log\left((\widehat{r}_{max} \vee r_{max})/\epsilon\right)\right)$, we choose δ, τ, η to satisfy $\delta = \mathcal{O}(\epsilon)$, $\tau = \Omega\left(\epsilon/\log|\mathcal{A}|\right)$, $\eta \leq (1-\gamma)/\tau$. Now, set $\mathbb{N}_{II}$ to be $\{n \mid n > \frac{1}{\eta\tau}\log\left(\frac{C_1(\gamma+2)}{\epsilon}\right), n \in \mathbb{N}\}$. Then $\mathfrak{R}_{II} \leq 4\epsilon(K-1)$.*

It is straightforward that sublinear $\mathfrak{R}_{II}$ can realized with any $\epsilon = \mathcal{O}((K-1)^{\alpha-1})$ for any $\alpha \in [0,1)$, which suggests a tighter upper bound of the $\mathfrak{R}_{II}$ requires a larger ϵ. Proposition 1 suggests that a tighter upper bound of the $\mathfrak{R}_{II}$ requires a smaller ϵ, which subsequently requires a larger $\Delta_\pi \in \mathbb{N}_{II}$. The hyperparameter conditions in Proposition 1 can be found in Lemma 1 and 2 in Appendix D.3.

4.1.2 $\mathfrak{R}_I$ analysis

We now present incorporating environment tempo $B(\Delta_\pi)$ into the upper bound of $\mathfrak{R}_I$ using the well-established non-stationary adaptation technique, Sliding Window regularized Least Squares Estimator (`SW-LSE`) [18–20], as a MDP forecaster. The tractability of `SW-LSE` algorithm allows it to establish an upper bound on model predictions errors $\iota_H^K, \bar{\iota}_\infty^K$ using environment tempo extracted from the past w observed trajectories, leading to sublinear $\mathfrak{R}_I$ as demonstrated in the following theorem.

Theorem 2 (Dynamic regret $\mathfrak{R}_I$ when f = `SW-LSE`)**.** *For given $p \in (0,1)$, if the exploration bonus constant β and regularization parameter λ satisfy $\beta = \Omega(|\mathcal{S}|H\sqrt{\log(H/p)})$, $\lambda \geq 1$, then the $\mathfrak{R}_I$ is bounded with probability $1-p$,*

$$\mathfrak{R}_I \leq C_I[B(\Delta_\pi)] \cdot w + C_k \cdot \sqrt{\frac{1}{w}\log\left(1 + \frac{H}{\lambda}w\right)}$$

where $C_I[B(\Delta_\pi)] = (1/(1-\gamma) + H) \cdot B_r(\Delta_\pi) + (1 + H\hat{r}_{max})\gamma/(1-\gamma) \cdot B_p(\Delta_\pi)$, and C_k is a constant on the order of $\mathcal{O}(K)$.

For the brief sketch of how `SW-LSE` takes out environment tempo as a upperbound, we outline that the model prediction errors are upperbounded by two forecaster errors between $P_{(k+1)}$ and $\widehat{P}_{(k+1)}$, as well as $R_{(k+1)}$ and $\widehat{R}_{(k+1)}$, along with the visitation count $n_{(k)}(s,a)$. Then `SW-LSE` algorithm facilitates $\widehat{P}_{(k+1)}, \widehat{R}_{(k+1)}$ be expressed in a closed form linear combinations of w past estimated values $\widetilde{P}, \widetilde{R}$. Finally, employing Cauchy inequality and triangle inequality derive two forecasting errors to be upperbounded by the environment tempo. For the final step before obtaining the optimal Δ_π^*, we compute the $\mathbb{N}_\pi$ that guarantees the sublinear $\mathfrak{R}_I$.

Proposition 2 (Δ_π bounds for sublinear $\mathfrak{R}_I$)**.** *Denote $B(1)$ as environment tempo for one policy iteration update. If environment satisfies $B_r(1) + B_p(1)\hat{r}_{max}/(1-\gamma) = o(K)$ and we choose $w = \mathcal{O}((K-1)^{2/3}/(C_I[B(\Delta_\pi)])^{2/3})$ and set $\mathbb{N}_I$ to be $\{n \mid n < K,\ n \in \mathbb{N}\}$, then $\mathfrak{R}_I$ is upperbound as $\mathfrak{R}_I = \mathcal{O}\left(C_I[B(\Delta_\pi)]^{1/3}(K-1)^{2/3}\sqrt{\log\left((K-1)/C_I[B(\Delta_\pi)]\right)}\right)$ and also satisfies sublinear upper bound.*

The upperbound on environment tempo for one policy update $B(1)$ in proposition 2 aligns with our expectations. his result is in line with the understanding that dedicating an excessively long time to a single iteration might not allow for effective policy approximation, thereby hindering the achievement of a sublinear dynamic regret. Moreover, our insight that a larger environment tempo prompts the MDP forecaster to consider a shorter past reference length, aiming to mitigate forecasting uncertainty, is consistent with the condition involving w stated in Proposition 2.

4.1.3 Optimal tempo Δ_π^*

So far, we show the dynamic regret of `ProST` is composed of two dynamic regrets $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$ and each of those are characterized by environment tempo and the agent tempo, respectively. Now, as we mentioned, we present that the optimal tempo Δ_π^* strikes a balance between the environment tempo and the agent tempo, since $\mathfrak{R}_I$ is non-decreasing function of Δ_π and $\mathfrak{R}_{II}$ is non-increasing function of Δ_π. Aligning with $\mathbb{N}_I, \mathbb{N}_{II}$, we now compute a minimizer of $\mathfrak{R}_I + \mathfrak{R}_{II}$ by taking its first derivative. We denote $k_{\texttt{Env}}$, $k_{\texttt{Agent}}$ as a constant occurs when taking first derivative of $\mathfrak{R}_I, \mathfrak{R}_{II}$, respectively. following proposition shows the optimal tempo Δ_π^* depends on the environment's drifting constants (Assumption 1).

Theorem 3 (Optimal tempo Δ_π^*)**.** *Let* $k_{Env} = (\alpha_r \vee \alpha_p)^2 C_I[B(1)]$, $k_{Agent} = \log(1/(1-\eta\tau))C_1(K-1)(\gamma+2)$ *where comes from* $\mathbb{N}_{II}$*. Then* Δ_π^* *depends on the environment's drifting constants ;* ***case1****:* $\alpha_r \vee \alpha_p = 0$*,* ***case2****:* $\alpha_r \vee \alpha_p = 1$*,* ***case3****:* $0 < \alpha_r \vee \alpha_p < 1$*,* ***case4****:* $\alpha_r \vee \alpha_p > 1$*.*

- *Case1:* $\Delta_\pi^* = \infty$*, Case2:* $\Delta_\pi^* = \log_{1-\eta\gamma}(k_{Env}/k_{Agent}) + 1$
- *Case3 & 4:* $\Delta_\pi^* = \exp\left(-W\left[-\frac{\log(1-\eta\tau)}{\max(\alpha_r,\alpha_p)-1}\right]\right)$ *if* $k_{Agent} = (1-\eta\tau)k_{Env}$*.*

4.2 Improve MDP forecaster

Determining the optimal tempo by taking a minimum of the upperbound of $\mathfrak{R}_I + \mathfrak{R}_{II}$ still leave room for an improvement that tighter upperbound of $\mathfrak{R}_I, \mathfrak{R}_{II}$ provides much accurate optimal tempo. On the shed of Proposition 1, we focus on the Q approximation gap δ to provide tighter $\mathfrak{R}_I + \mathfrak{R}_{II}$. It is important to note that the factor δ arises not only from the finite sample trajectories as discussed in [21], but also the forecasting error between $\mathcal{M}_{(k+1)}$ and $\widehat{\mathcal{M}}_{(k+1)}$. This is clear that the MDP forecaster establishes a lower bound of δ denoted as $\delta_{\min}$, which in turn sets a lowerbound on ϵ and consequently on $\mathfrak{R}_I$. This inspection highlights that the MDP forecaster serves as a common factor that controls both $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$, and further investigation on improving forecaster accuracy is necessary for tighter $\mathfrak{R}_I + \mathfrak{R}_{II}$.

Our approach to devising a precise MDP forecaster is to, instead of *selecting* the past reference length w as indicated in Proposition 2, set $w = k$, implying the utilization of all past observations. However, we address this by solving an additional optimization problem, resulting in a tighter $\mathfrak{R}_I$. We propose a method that adaptively assigns the different weights $q \in \mathbb{R}_+^k$ on all of the past observed non-stationary variables until time t_k, which reduces the burden of choosing w. Hence, we introduce further analysis to demonstrate the tighter $\mathfrak{R}_I$ through the utilization of the Weighted regularized Least Squares Estimator (`W-LSE`) [22]. Unlike `SW-LSE`, `W-LSE` doesn't necessitate a predefined selection of w, instead, it engages in a joint optimization procedure involving data weights q and future model $\widehat{P}_{(k+1)}, \widehat{R}_{(k+1)}$. Before, we define forecasting reward model error as $\Delta_k^r(s,a) = \left|\left(R_{(k+1)} - \widehat{R}_{(k+1)}\right)(s,a)\right|$, forecasting transition probability model error as $\Delta_k^p(s,a) = \left\|\left(P_{(k+1)} - \widehat{P}_{(k+1)}\right)(\cdot \mid s,a)\right\|_1$.

Theorem 4 ($\mathfrak{R}_I$ upper bound when f=`W-LSE`)**.** *Set exploration bonus* $\Gamma_{(k)}(s,a) = \frac{1}{2}\Delta_k^r(s,a) + \frac{\gamma\tilde{r}_{max}}{2(1-\gamma)}\Delta_k^p(s,a)$*, then*

$$\mathfrak{R}_I \le \left(4H + \frac{2\gamma|\mathcal{S}|}{1-\gamma}\left(\frac{1}{1-\gamma} + H\right)\right)\left(\frac{1}{2}\sum_{k=1}^{K-1}\Delta_k^r(s,a) + \frac{\gamma\tilde{r}_{max}}{2(1-\gamma)}\sum_{k=1}^{K-1}\Delta_k^p(s,a)\right)$$

Now, Remark 1 demonstrates that the upper bound of $\bar{\iota}_\infty^K, -\iota_H^K$ becomes tighter by solving a joint optimization problem, thus leads to the tighter upperbound of $\mathfrak{R}_\mathcal{I}$.

Remark 1 (Tighter $\mathfrak{R}_\mathcal{I}$ upper bound when f = `W-LSE`)**.** *If the joint optimization problem of* `W-LSE` *feasible, then the optimal data weight* q^* *provides a tighter upper bound for* Δ_k^r, Δ_k^p *in comparison to* `SW-LSE`*, consequently leading to a tighter* $\mathfrak{R}_\mathcal{I}$ *upper bound. We provide* $\bar{\iota}_\infty^K, -\iota_H^K$ *is upperbounded by* Δ_k^r, Δ_k^p *in Lemma 4 and 6 (Appendix D.3).*

4.3 ProST-G

The theoretical analysis outlined above serves as motivation to empirically investigate two key points: firstly, the existence of an optimal training time; secondly, the role of the MDP forecaster's contribution to the `ProST` framework's overall performance. To address these questions, We propose a practical instance, `ProST-G`, that particularly extends investigation in section 4.2. `ProST-G` optimizes a policy with soft actor-critic (SAC) algorithm [23], utilizes integrated autoregressive integrated moving average (ARIMA) model for the proactive forecaster f, and uses a bootstrap ensemble of dynamic models where each model is a probabilistic neural network for the model predictor g. We further discuss specific details of `ProST-G` in Appendix F.3 and in Algorithm 3

5 Experiments

We evaluated ProST-G with four baselines in three Mujoco environments each with five different non-stationary speeds and two non-stationary datasets.

(1) Environments: Non-stationary desired posture. We made the rewards in the three environments non-stationary by altering the agent's desired directions. Forward reward R_t^f changes as $R_t^f = o_t \cdot \bar{R}_t^f$, where $\bar{R}_f$ is the original reward from the Mujoco environment. Non-stationary variable o_k is generated from sin function with five different speeds and from the real data A, B, and we measure time elapsing variation budget by $\sum_{k=1}^{K-1} |o_{k+1} - o_k|$. Further details of the environment settings can be found in Appendix D.1.1.

(2) Benchmark methods. Four baselines are chosen to empirically support our second question: the significance of the forecaster. **MBPO** is the state-of-the-art model-based policy optimization [24]. **Pro-OLS** is a policy gradient algorithm that predicts the future performance and optimizes the predicted performance of the future episode [7]. **ONPG** is an adaptive algorithm that performs a purely online optimization by fine-tuning the existing policy using only the trajectory observed online [8]. **FTRL** is an adaptive algorithm that performs follow-the-regularized-leader optimization by maximizing the performance on all previous trajectories [9].

6 Discussions

6.1 Performance compare

The summary of the experimental results is presented in Table 1. The table summarizes the average return over the last 10 episodes during the training procedure. We have illustrated the complete training results in the Appendix E.3. In most cases, ProST-G outperforms MBPO in terms of rewards, highlighting the adaptability of our ProST framework to dynamic environments. Furthermore, except for data A and B, ProST-G consistently outperforms the other three baselines. This supports our motivation of using the proactive model-based method for higher adaptability in non-stationary environments compared to state-of-the-art model-free algorithms (Pro-OLS, ONPG, FTRL). We elaborate on the training details in Appendix E.2.

Table 1: Average reward returns

Speed	$B(G)$	**Swimmer-v2**					**Halfcheetah-v2**					**Hopper-v2**				
		Pro-OLS	ONPG	FTML	MBPO	ProST-G	Pro-OLS	ONPG	FTML	MBPO	ProST-G	Pro-OLS	ONPG	FTML	MBPO	ProST-G
1	16.14	-0.40	-0.26	-0.08	-0.08	**0.57**	-83.79	-85.33	-85.17	-24.89	**-19.69**	**98.38**	95.39	97.18	92.88	92.77
2	32.15	0.20	-0.12	0.14	-0.01	**1.04**	-83.79	-85.63	-86.46	-22.19	**-20.21**	98.78	97.34	**99.02**	96.55	98.13
3	47.86	-0.13	0.05	-0.15	-0.64	**1.52**	-83.27	-85.97	-86.26	-21.65	**-21.04**	97.70	98.18	98.60	95.08	**100.42**
4	63.14	-0.22	-0.09	-0.11	-0.04	**2.01**	-82.92	-84.37	-85.11	-21.40	**-19.55**	98.89	97.43	97.94	97.86	**100.68**
5	77.88	-0.23	-0.42	-0.27	0.10	**2.81**	-84.73	-85.42	-87.02	**-20.50**	-20.52	97.63	99.64	99.40	96.86	**102.48**
A	8.34	1.46	2.10	**2.37**	-0.08	0.57	-76.67	-85.38	-83.83	-40.67	**83.74**	104.72	**118.97**	115.21	100.29	111.36
B	4.68	**1.79**	-0.72	-1.20	0.19	0.20	-80.46	-86.96	-85.59	-29.28	**76.56**	80.83	**131.23**	110.09	100.29	127.74

6.2 Ablation study

An ablation study was conducted on the two aforementioned questions. The following results support our inspection of section 4.2 and provide strong grounds for the Theorem 3.

Optimal Δ_π^*. The experiments are performed over five different policy training times $\Delta_\pi = [1, 2, 3, 4, 5]$, aligned with SAC's policy iterations $[38, 76, 114, 152, 190]$, under a fixed environment speed. Different from theoretical analysis, we have equated $\Delta_t = 1$ with $G = 38$. We generate$o_k = sin(2\pi\Delta_\pi k/37)$ which satisfies Assumption 1 (see Appendix E.1). The shaded area of Figure 3 (a), (b-1), (b-2) are 95 % confidence area among three different noise bounds $[0.01, 0.02, 0.03]$ in o_k. Figure 3-(a) shows $\Delta_t = 4$, $G = 152$ is close to the optimal G^* among five different choices.

Function f, g. We investigate the effect of forecaster f's accuracy on the framework using two distinct functions: ARIMA and a simple average (SA) model, each tested with three different ws. Figure 3-(b-1) shows the average rewards of the SA model with the $w = [3, 5, 7]$ and ARIMA model (four solid lines). The shaded area is 95 % the confidence area among 4 different speeds [1,2,3,4]. Figure 3-(b-2) shows the corresponding model error. Also, we investigate the effect of different model predictors g by comparing MBPO (reactive-model) and ProST-G with f =ARIMA

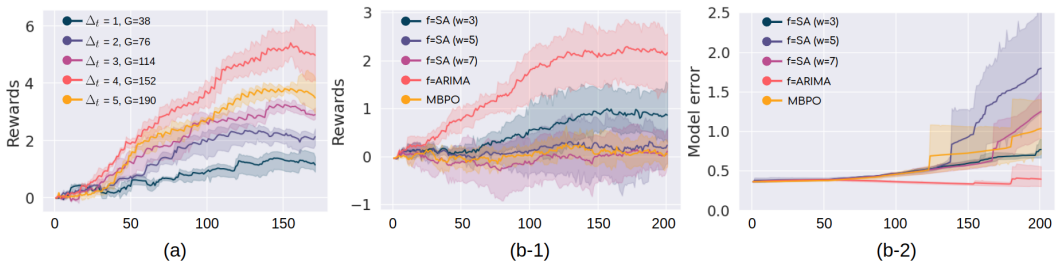

Figure 3: (a) Optimal $\Delta_\pi^\star$. (b-1) Different forecaster f (ARIMA, SA). (b-2) The Mean squared Error (MSE) model loss of four `ProST-G` with different forecasters(ARIMA and three SA) and the MBPO. x-axis are all episodes.

(proactive-model) in Figure 3-(b-2). The high returns from `ProST-G` with f = ARIMA, compared to those from MBPO empirically supports that the forecasting component of **ProST** framework can provide good adaptability to the baseline algorithm that is equipped with. Also, Figure 3-(b-1) and 3-(b-2) provide empirical evidence that f accuracy is contingent on the sliding window size, thereby impacting model accuracy and subsequently influencing the agent's performance.

7 Conclusion, Limitations, Future works

To the best of our knowledge, we first tackle the internal assumption, time synchronization, of non-stationary RL that holds back its primary motivation. To solve this issue, we newly introduce and focus on the tempo of adaptation in a non-stationary RL, and find optimal training time. In this paper, we propose a Proactively Synchronizing Tempo (ProST) framework, and two specific instances `ProST-T` and `ProST-G`. Our proposed method adjusts an agent's tempo to match the tempo of the environment to address non-stationarity through theoretical analysis and empirical evidence. In summary, we believe that the ProST framework provides a new avenue to implement reinforcement learning in the real world by incorporating the concept of adaptation tempo, but our work does not cover various factors of the real world.

One way to expand our work is learning a safe guarantee policy in a non-stationary RL [25–27] by considering the adaption tempo of constraint violations. Another way is finding a good tempo of the distribution correction in offline non-stationary RL, specifically how to adjust the relabeling function to offline data in a time-varying environment that is dependent on the tempo of the environment. [28, 29].

References

[1] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[2] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, L. Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Vedavyas Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy P. Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529:484–489, 2016.

[3] Gabriel Dulac-Arnold, Daniel Mankowitz, and Todd Hester. Challenges of real-world reinforcement learning. *arXiv preprint arXiv:1904.12901*, 2019.

[4] Luisa Zintgraf, Kyriacos Shiarlis, Maximilian Igl, Sebastian Schulze, Yarin Gal, Katja Hofmann, and Shimon Whiteson. Varibad: A very good method for bayes-adaptive deep rl via meta-learning. *arXiv preprint arXiv:1910.08348*, 2019.

[5] Zhuangdi Zhu, Kaixiang Lin, Anil K Jain, and Jiayu Zhou. Transfer learning in deep reinforcement learning: A survey. *arXiv preprint arXiv:2009.07888*, 2020.

[6] Sindhu Padakandla. A survey of reinforcement learning algorithms for dynamically varying environments. *ACM Computing Surveys (CSUR)*, 54(6):1–25, 2021.

[7] Yash Chandak, Georgios Theocharous, Shiv Shankar, Martha White, Sridhar Mahadevan, and Philip Thomas. Optimizing for the future in non-stationary mdps. In *International Conference on Machine Learning*, pages 1414–1425. PMLR, 2020.

[8] Maruan Al-Shedivat, Trapit Bansal, Yuri Burda, Ilya Sutskever, Igor Mordatch, and Pieter Abbeel. Continuous adaptation via meta-learning in nonstationary and competitive environments. *arXiv preprint arXiv:1710.03641*, 2017.

[9] Chelsea Finn, Aravind Rajeswaran, Sham Kakade, and Sergey Levine. Online meta-learning. In *International Conference on Machine Learning*, pages 1920–1930. PMLR, 2019.

[10] Yuhao Ding, Ming Jin, and Javad Lavaei. Non-stationary risk-sensitive reinforcement learning: Near-optimal dynamic regret, adaptive detection, and separation design. *arXiv preprint arXiv:2211.10815*, 2022.

[11] STEVEN J BRADTKE. Linear least-squares algorithms for temporal difference learning. *Machine Learning*, 22:33–57, 1996.

[12] Yonatan Gur, Assaf J. Zeevi, and Omar Besbes. Stochastic multi-armed-bandit problem with non-stationary rewards. In *NIPS*, 2014.

[13] Xiaoyu Chen, Xiangming Zhu, Yufeng Zheng, Pushi Zhang, Li Zhao, Wenxue Cheng, Peng Cheng, Yongqiang Xiong, Tao Qin, Jianyu Chen, et al. An adaptive deep rl method for non-stationary environments with piecewise stable context. *arXiv preprint arXiv:2212.12735*, 2022.

[14] Biwei Huang, Fan Feng, Chaochao Lu, Sara Magliacane, and Kun Zhang. Adarl: What, where, and how to adapt in transfer reinforcement learning. *arXiv preprint arXiv:2107.02729*, 2021.

[15] Fan Feng, Biwei Huang, Kun Zhang, and Sara Magliacane. Factored adaptation for non-stationary reinforcement learning. *arXiv preprint arXiv:2203.16582*, 2022.

[16] Jeongyeol Kwon, Yonathan Efroni, Constantine Caramanis, and Shie Mannor. Rl for latent mdps: Regret guarantees and a lower bound. *Advances in Neural Information Processing Systems*, 34:24523–24534, 2021.

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

[18] Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Learning to optimize under non-stationarity. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1079–1087. PMLR, 2019.

[19] Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Hedging the drift: Learning to optimize under nonstationarity. *Management Science*, 68(3):1696–1713, 2022.

[20] Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Reinforcement learning for non-stationary markov decision processes: The blessing of (more) optimism. In *International Conference on Machine Learning*, pages 1843–1854. PMLR, 2020.

[21] Shicong Cen, Chen Cheng, Yuxin Chen, Yuting Wei, and Yuejie Chi. Fast global convergence of natural policy gradient methods with entropy regularization. *Operations Research*, 70(4): 2563–2578, 2022.

[22] Vitaly Kuznetsov and Mehryar Mohri. Theory and algorithms for forecasting time series. *arXiv preprint arXiv:1803.05814*, 2018.

[23] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. PMLR, 2018.

[24] Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32, 2019.

[25] Ming Jin and Javad Lavaei. Stability-certified reinforcement learning: A control-theoretic perspective. *IEEE Access*, 8:229086–229100, 2020.

[26] Vanshaj Khattar, Yuhao Ding, Javad Lavaei, and Ming Jin. Provable guarantees for meta-safe reinforcement learning.

[27] Samuel Pfrommer, Tanmay Gautam, Alec Zhou, and Somayeh Sojoudi. Safe reinforcement learning with chance-constrained model predictive control. In *Learning for Dynamics and Control Conference*, pages 291–303. PMLR, 2022.

[28] Jongmin Lee, Cosmin Paduraru, Daniel J Mankowitz, Nicolas Heess, Doina Precup, Kee-Eung Kim, and Arthur Guez. Coptidice: Offline constrained reinforcement learning via stationary distribution correction estimation. *arXiv preprint arXiv:2204.08957*, 2022.

[29] Jongmin Lee, Wonseok Jeon, Byungjun Lee, Joelle Pineau, and Kee-Eung Kim. Optidice: Offline policy optimization via stationary distribution correction estimation. In *International Conference on Machine Learning*. PMLR.

[30] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

[31] Weichao Mao, Kaiqing Zhang, Ruihao Zhu, David Simchi-Levi, and Tamer Başar. Model-free non-stationary rl: Near-optimal regret and applications in multi-agent rl and inventory control. *arXiv preprint arXiv:2010.03161*, 2020.

[32] Yuhao Ding and Javad Lavaei. Provably efficient primal-dual reinforcement learning for cmdps with non-stationary objectives and constraints. *arXiv preprint arXiv:2201.11965*, 2022.

[33] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas Kirkeby Fidjeland, Georg Ostrovski, Stig Petersen, Charlie Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015.

[34] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[35] Wen Sun, Geoffrey J Gordon, Byron Boots, and J Bagnell. Dual policy iteration. *Advances in Neural Information Processing Systems*, 31, 2018.

[36] Yuping Luo, Huazhe Xu, Yuanzhi Li, Yuandong Tian, Trevor Darrell, and Tengyu Ma. Algorithmic framework for model-based deep reinforcement learning with theoretical guarantees. *arXiv preprint arXiv:1807.03858*, 2018.

[37] Yingjie Fei, Zhuoran Yang, Zhaoran Wang, and Qiaomin Xie. Dynamic regret of policy optimization in non-stationary environments. *Advances in Neural Information Processing Systems*, 33:6743–6754, 2020.

[38] Yash Chandak, Scott Jordan, Georgios Theocharous, Martha White, and Philip S Thomas. Towards safe policy improvement for non-stationary mdps. *Advances in Neural Information Processing Systems*, 33:9156–9168, 2020.

[39] Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *The Journal of Machine Learning Research*, 22(1):4431–4506, 2021.

[40] Gen Li, Yuting Wei, Yuejie Chi, Yuantao Gu, and Yuxin Chen. Breaking the sample size barrier in model-based reinforcement learning with a generative model. *Advances in neural information processing systems*, 33:12861–12872, 2020.

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

A Details on Introduction

A.1 Experimental motivation

1. Environment details of 2D goal reacher.

- State space: $\mathcal{S} = \mathbb{R}^2$. For $(x, y) \in \mathcal{S}, |x| \leq 1, |y| \leq 1$.
- Action space: $\mathcal{A} = \{(\cos(\pi/4 * k), \sin(\pi/4 * k)) \mid k = 0, 1, ..7\}$ ($|\mathcal{A}| = 8$)
- Reward function: if agent's state is in the Goal box, then receives +6. Otherwise, receives -0.5 rewards for every step.
- Transition probability: $s_{h+1} = s_h + a_h \cdot \epsilon$ where s_{h+1} is next state, s_h is current state, a_h is the current action, $\epsilon \in \mathbb{R}^2, ||\epsilon||_2 = 1$ provides a stocasticity to the environment.
- Horizon length: $H = 13$
- Discounting factor: $\gamma = 0.99$
- Grid size: 10
- Goal box : The center coordinate of the time-varying goal box (x_g, y_g)= $(0.9\cos(2\pi * k/2500), 0.9\sin(2\pi * k/2500))$ changes for episode $k \in [5000]$. The box width and height are 0.05

2. Experiment details.

To motivate our proposed meta-framework as a simple experiment, we used Q-learning as a component A of our meta-algorithm to update the policy. The three baselines (ProOLS, ONPG, FTML) of Figure 1-(c) were trained with four learning rates $\eta = \{1e-3, 3e-3, 5e-3, 7e-3\}$, entropy regularized parameter $\tau = 0.1$, and the shaded area of the three baselines is 95 % confidence area among 4 different learning rates. The `PTM-T` was trained with model rollout length $\widehat{H} = \{50, 60\}$, policy update iteration number $G = \{10, 50\}$, entropy regularized parameter $\tau = 0.1$, Q learning update parameter $\alpha_Q = \{0.7, 0.9, 0.99\}$ and the learning rate $\eta = 1e-3$. The shaded area of `PTM-T` is 95 % confidence area among the 12 different cases above. All four algorithms have the same structure of the agent's policy network.

B Related works

Existing methods for non-stationary environments can be grouped into three schools of thought: 1) shoehorning: directly using established frameworks for stationary MDPs, assuming no extra mechanisms are needed since non-stationarity already exists in standard RL due to policy updates; 2) model-based policy updates: updating models with new data, using short rollouts to prevent model exploitation [24, 30], online model updates, or through latent factor identification [4, 13–16]; and 3) anticipating future changes by forecasting policy gradients or value functions [7, 31, 20, 10, 32].

The advantage of the model-free method is its computational efficiency, allowing for direct learning of complex policies from raw data [33, 34], while the advantage of the model-based method is its data efficiency, allowing one to learn fast by learning how the environment works [35, 36]. However, both advantages are weakened in non-stationary environments since the optimizing non-stationary loss function induced by time-varying data distribution makes the model-free method challenging to adaptively obtain the optimal policy [37, 38] and the model-based method challenging to estimate accurate non-stationary models [20, 10].

Model-free method in non-stationary RL. [8] uses meta-learning among the training tasks to find initial hyperparameters of the policy networks that can be quickly fine-tuned when facing testing tasks that have not been encountered before. However, access to a prior distribution of training tasks is not available in real-world problems. To mitigate this issue, [9] proposed the Follow-The-Meta-Leader (FTML) algorithm that continuously improves an initialization of parameters for non-stationary input data. However, it internally entails a lag when tracking optimal policy as it maximizes the current performance over all the past samples uniformly. To alleviate the lag problem, [7, 38] focus on directly forecasting the non-stationary performance gradient to adapt the time-varying optimal policies. However, it still has problems of showing empirical analysis on bandit settings or a low-dimensional environment and lack of theoretical analysis which provides a bound on the adapted

policy's performance. [31] proposed adaptive Q-learning with a restart strategy and established its near-optimal dynamic regret bound. In addition, [37] proposed two model-free policy optimization algorithms based on the restart strategy and showed that dynamic regret satisfies polynomial space and time complexities. However, those provable model-free methods [31, 37] still lack empirical evidence and adaptability in complex environments. Furthermore, since the agent can only execute a policy in a fixed environment once due to the non-stationary of the environments, most existing model-free methods only update the policy once for each environment, which prevents the tracking of the time-varying optimal policies.

Model-based method in non-stationary RL. [14] learns the model change factors and their representation in heterogeneous domains with varying reward functions and dynamics. However, it is restricted to use in non-stationary environments, that is, applicable only for constant change factors or the domain adaptation setting. [4] proposed a Bayesian optimal learning policy algorithm by conditioning the action on both states and latent vectors that capture the agent's uncertainty in the environment. Also, [15] brings insights from recent causality research to model non-stationarity as latent change factors across different environments, and learn policy conditioning on latent factors of the causal graphs. However, learning an optimal policy conditioning on the latent states [4, 13–16] makes the theoretical analysis intractable, namely we cannot disentangle the error that comes from the accurate mapping between latent states and environment states and the optimization error that both simultaneously affect the performance. Some recent works [20, 10, 32] proposed model-based algorithms with a provable guarantee, but their algorithms are not scalable for complex environments and lack empirical evaluation for complex environments.

C Details on Problem statement and Notations

C.1 Details on Notations

Environment Interaction. At episode k, the agent starts from an initial state $s_0^k \sim \rho$. At step $h \in [H]$ of the episode k, the agent takes an action $a_h^k = \pi^k(s_h^k)$ from the current state s_h^k. The agent then receives the reward $r_h^k \sim R^{(k)}(s_h^k, a_h^k)$ and moves to the next state $s_{h+1}^k \sim P^{(k)}(s_{h+1}^k | s_h^k, a_h^k)$. The episode k ends when the agent reaches s_{H+1}^k.

Future MDP $\widehat{\mathcal{M}}_{t_{k+1}}$. Our work starts from creating one-episode ahead MDP $\widehat{\mathcal{M}}_{t_{k+1}}$ based on the observed data from p lastest MDPs $\{\mathcal{M}_{t_{k-p+1}}, ..., \mathcal{M}_{t_k}\}$ when the agent is stated in episode k. We define $\widehat{\mathcal{M}}_{t_{k+1}} \coloneqq \langle \mathcal{S}, \mathcal{A}, H, \widehat{P}^{t_{k+1}}, \widehat{R}^{t_{k+1}}, \gamma \rangle$ where $\widehat{\mathcal{P}}^{k+k_f}, \widehat{R}^{k+k_f}$ are *forecasted* future transition probability and reward function, respectively. The agent also interacts with the created future MDP $\widehat{\mathcal{M}}_{t_{k+1}}$ in the same way as it did with the original MDP $\mathcal{M}_{t_k}$. We denote state, action, and policy in $\widehat{\mathcal{M}}_{t_{k+1}}$ as $\widehat{s}_h^{t_{k+1}}, \widehat{a}_h^{t_{k+1}}, \widehat{\pi}^{t_{k+1}}$. We elaborate our main methodology in Section 3

State value, state-action value function. For any given policy π and the MDP $\mathcal{M}_k$, We denote the state value function at episode k as $V^{\pi,k} : \mathcal{S} \to \mathbb{R}$ and the state action value function at episode k as $Q^{\pi,k} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. We define

$$V^{\pi,k}(s) \coloneqq \mathbb{E}_{\mathcal{M}_{(k)},\pi}\left[\sum_{h=0}^{H-1} \gamma^h r_h^k \,\middle|\, s_0^k = s\right],$$

$$Q^{\pi,k}(s,a) \coloneqq \mathbb{E}_{\mathcal{M}_{(k)},\pi}\left[\sum_{h=0}^{H-1} \gamma^h r_h^k \,\middle|\, s_0^k = s,\ a_0^k = a\right].$$

Also, given the future MDP $\widehat{\mathcal{M}}_{(k+1)}$, we denote the *forecasted* state value as $\widehat{V}^{\pi,k+1}(s) : \mathcal{S} \to \mathbb{R}$ and *forecasted* state-action value as $\widehat{Q}^{\pi,k+1} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. We define

$$\widehat{V}^{\pi,k+1}(s) \coloneqq \mathbb{E}_{\widehat{\mathcal{M}}_{k+1},\pi}\left[\sum_{h=0}^{H-1} \gamma^h \widehat{r}_h^{k+1} \,\middle|\, \widehat{s}_0^{k+1} = s\right],$$

$$\widehat{Q}^{\pi,k+1}(s,a) \coloneqq \mathbb{E}_{\widehat{\mathcal{M}}_{k+1},\pi}\left[\sum_{h=0}^{H-1} \gamma^h \widehat{r}_h^{k+1} \,\middle|\, \widehat{s}_0^{k+1} = s,\ \widehat{a}_0^{k+1} = a\right].$$

Dynamic regret. Aside from the stationary MDPs, the agent aims to maximize the cumulative expected reward throughout the K episodes by adopting a sequence of policies $\{\pi^k\}^{k=1:K}$. In non-

stationary MDPs, the optimality of the policy is evaluated in terms of dynamic regret $\mathfrak{R}\left(\{\pi^k\}, K\right)$.

$$\mathfrak{R}\left(\{\pi^k\}_{1:K}, K\right) \coloneqq \sum_{k=1}^{K} \left(V^{*,k}(\rho) - V^{\pi^k,k}(\rho)\right) \tag{C.1}$$

where we denote $V^{*,k}(= V^{\pi^{*,k},k})$ as the optimal state value function with the optimal policy $\pi^{*,k}$ at the episode k and $V^{\pi^k,k}$ as the state value with agent's k^{th} episode's policy π^k. Dynamic regret is a stronger evaluation than the standard static regret which considers the optimality of the single policy over the entire episodes.

Model prediction error. To measure how well our meta-function predicts the future environment, we define two different *model prediction error*s $\iota_\infty^{k+1}, \iota_h^{k+1} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, which denote the bellman equation error when using $\widehat{V}, \widehat{Q}$ estimated in the future MDP instead of the true V, Q function:

$$\iota_\infty^{k+1}(s,a) \coloneqq \left(R^{k+1} + \gamma P^{k+1} \widehat{V}_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1}\right)(s,a), \tag{C.2}$$

$$\iota_h^{k+1}(s,a) \coloneqq \left(R^{k+1} + \gamma P^{k+1} \widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1} - \widehat{Q}_h^{\widehat{\pi}^{k+1},k+1}\right)(s,a) \tag{C.3}$$

State value, state-action value function of step h. We denote the state value function, state action value function for any policy π at *step* h of the episode k as $V_h^{\pi,k}, Q_h^{\pi,k}$ respectively. We define

$$V_h^{\pi,k}(s) \coloneqq \mathbb{E}_{\mathcal{M}_{(k)},\pi}\Big[\sum_{i=h}^{H-1} \gamma^{i-h} r_i^k \mid s_h^k = s\Big],$$

$$Q_h^{\pi,k}(s,a) \coloneqq \mathbb{E}_{\mathcal{M}_{(k)},\pi}\Big[\sum_{i=h}^{H-1} \gamma^{i-h} r_i^k \mid s_h^k = s, a_h^k = a\Big].$$

Then, the corresponding Bellman equation is

$$Q_h^{\pi,k}(s,a) = \left(R^{(k)} + \gamma P^{(k)} V_{h+1}^{\pi,k}\right)(s,a), \;\; V_h^{\pi,k}(s) = \left\langle Q_h^{\pi,k}, \pi \right\rangle_{\mathcal{A}}, \;\; V_H^{k,\pi}(s) = 0 \;\forall s \in \mathcal{S} \tag{C.4}$$

where we define $P^{(k)} V_{h+1}(s,a) \coloneqq \mathbb{E}_{s' \sim P^{(k)}(\cdot|s,a))}[V_{h+1}(s')]$.

We denote $V_h^{*,k}(s) = V_h^{\pi^*,k}(s))$ as the optimal state value function of step h of episode k. We omit the underscript of h when $h = 0$, that is, $V^{\pi,k} = V_0^{\pi,k}, \; Q^{\pi,k} = Q_0^{\pi,k}$. Then, the corresponding bellman optimal equation is

$$Q_h^{*,k}(s,a) = \left(R^{(k)} + \gamma P^{(k)} V_{h+1}^{*,k}\right)(s,a), \;\; V_h^{*,k}(s) = \left\langle Q_h^{*,k}, \pi^* \right\rangle_{\mathcal{A}}, \;\; \pi^* = \max_a Q_h^{*,k}(s,a), \tag{C.5}$$

We also denote k_f-ahead*forecasted* state value of step h of episode$k$ as $\widehat{V}_h^{\pi,k+1}$ and k_f-ahead *forecasted* state-action value of step h of episode k as $\widehat{Q}_h^{\pi,k+1}$ in a forecasted MDP $\widehat{\mathcal{M}}_{k+1}$. We define

$$\widehat{V}_h^{\pi,k+1}(s) \coloneqq \mathbb{E}_{\widehat{\mathcal{M}}_{k+1},\pi}\Big[\sum_{i=h}^{H-1} \gamma^{i-h} \widehat{r}_i^{k+1} \mid \hat{s}_h^{k+1} = s\Big], \tag{C.6}$$

$$\widehat{Q}_h^{\pi,k+1}(s,a) \coloneqq \mathbb{E}_{\widehat{\mathcal{M}}_{k+1},\pi}\Big[\sum_{i=h}^{H-1} \gamma^{i-h} \widehat{r}_i^{k+1} \mid \hat{s}_h^{k+1} = s, \hat{a}_h^{k+1} = a\Big]. \tag{C.7}$$

Then, the bellman equation is given by

$$\widehat{Q}_h^{\pi,k+1}(s,a) = \left(\widehat{R}^{(k+1)} + \gamma \widehat{P}^{(k+1)} \widehat{V}_{h+1}^{\pi,k+1}\right)(s,a), \;\; \widehat{V}_h^{\pi,k+1}(s) = \left\langle \widehat{Q}_h^{\pi,k+1}, \pi \right\rangle_{\mathcal{A}}$$
$$\widehat{V}_H^{k+1,\pi}(s) = 0 \;\forall s \in \mathcal{S} \tag{C.8}$$

Let us also denote the *future* optimal policy of the *future* value function $\widehat{V}_0^{\pi,k+1}$ as $\widehat{\pi}^{*,k+1}$. Then the bellman optimal equation also holds for $\widehat{Q}_h^{\pi,k+1}(s), \widehat{V}_h^{\pi,k+1}(s)$ as follows.

$$\widehat{Q}_h^{*,k+1}(s,a) = \left(\widehat{R}^{(k+1)} + \gamma \widehat{P}^{(k+1)} \widehat{V}_{h+1}^{*,k+1}\right)(s,a), \;\; \widehat{V}_h^{*,k+1}(s) = \left\langle \widehat{Q}_h^{*,k+1}, \widehat{\pi}^* \right\rangle_{\mathcal{A}},$$
$$\widehat{\pi}^* = \max_a \widehat{Q}_h^{*,k+1}(s,a) \tag{C.9}$$

Unnormalized (discounted) occupancy measure. We define the unnormalized (discounted) occupancy measure $\nu^{\pi,k}_{s_0,a_0} \in \Delta_{1/(1-\gamma)}(\mathcal{S}\times\mathcal{A})$ at episode k for given policy π and an initial state and action s_0, a_0 as

$$\nu^{\pi,k}_{(s_0,a_0)}(s,a) \coloneqq \sum_{h=0}^{\infty} \gamma^h \mathbb{P}(s_h = s, a_h = a \mid s_0, a_0 \,;\, \pi, P^{(k)}) \quad , \quad \forall (s,a) \in \mathcal{S}\times\mathcal{A} \tag{C.10}$$

where $\mathbb{P}(s_h = s, a_h = a \mid s_0, a_0 \,;\, \pi, P^{(k)})$ is the probability of visiting (s,a) at step h when following policy π from s_0, a_0 with transition probability $P^{(k)}$.

We also define the unnormalized non-stationary (discounted) *forecasted* occupancy measure $\widehat{\nu}^{\pi,k+1}_{s_0} \in \Delta_{1/(1-\gamma)}(\mathcal{S}\times\mathcal{A})$ at episode k for given policy π and an initial state s_0, a_0,

$$\widehat{\nu}^{\pi,k+1}_{s_0,a_0}(s,a) \coloneqq \sum_{h=0}^{\infty} \gamma^h \mathbb{P}(s_h = s, a_h = a \mid s_0, a_0, \pi, \widehat{P}^{(k+1)}) \quad , \forall (s,a) \in \mathcal{S}\times\mathcal{A} \tag{C.11}$$

where probability is defined in a forecasted environment with $\widehat{P}^{(k+1)}$.

D Proof of theoretical analysis

D.1 preliminary for ProST-T and theoretical analysis

In this subsection, we elaborate the `ProST-T`'s environment setting and its meta-algorithm f, g in the following.

D.1.1 Environment setting

We consider the tabular environment to satisfy the following two points.

1. First, $P^{(k)}, R^{(k)}$ are represented by the inner products of the feature function $\phi : \mathcal{S}\times\mathcal{S}\times\mathcal{A} \to \mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|}, \varphi : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ and the non-stationary variables $o^p_{(k)}, o^r_{(k)} \in \mathcal{O}$, respectively, where $o^p_{(k)} : \mathcal{S}\times\mathcal{S}\times\mathcal{A} \to \mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|}$, $o^r_{(k)} : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$. That is, $P^{(k)} =< \phi, o^p_{(k)} >$, $R^{(k)} =< \varphi, o^r_{(k)} >$.
2. Second, the agent estimates $o^p_{(k)}, o^r_{(k)}$ rather than observe. More specifically, we consider the non-stationary variable set $\mathcal{O}$ to be the set $\{P\}^K_{k=1}, \{R\}^K_{k=1}$. The agent then tries to *estimate* o_k (denote $P^{(k)}$ as $o^p_{(k)}$ and $R^{(k)}$ as $o^r_{(k)}$) rather than observe it through its w lastest trajectories, where the assumption 2 does not necessarily hold in this setting. That is, the agent estimates $P^{(k)}$ by $\hat{o}^p_k$ and $R^{(k)}$ by $\hat{o}^r_k$ from observations of the w last trajectories, $\tau_{k-(w-1):k}$.

We elaborate the above two points as follows.

1. $P^{(k)}, R^{(k)}$ are inner products of ψ, φ and $o^p_{(k)}, o^r_{(k)}$.

Let us define a set of hot reward vectors over all states and the action space $\mathbb{1}_r \coloneqq \{\varphi^y \in \{0,1\}^{|\mathcal{S}||\mathcal{A}|} \mid \sum_{i=1}^{|\mathcal{S}||\mathcal{A}|} \varphi^y_i = 1\}$ and similarly define a set of hot transition probability vectors $\mathbb{1}_p \coloneqq \{\psi^y \in \{0,1\}^{|\mathcal{S}|^2|\mathcal{A}|} \mid \sum_{i=1}^{|\mathcal{S}|^2|\mathcal{A}|} \psi^y_i = 1\}$. Then define a one-to-one function φ, ψ that satisfies $\varphi : \mathcal{S}\times\mathcal{A} \to \mathbb{1}_r$, $\psi : \mathcal{S}\times\mathcal{S}\times\mathcal{A} \to \mathbb{1}_p$. Namely, $\varphi(s,a)(\psi(s',s,a))$ is a one-hot vector that the $(i)^{th}$ entry equals to 1. We denote the notation $\varphi^k_h = \varphi(s^k_h, a^k_h)$ for the observed (s^k_h, a^k_h) on the trajectory τ_k, and the same as $\psi^k_h = \psi(s^k_{h+1}, s^k_h, a^k_h)$.

Then, we set $\mathcal{O} = \{P^{(k)}, R^{(k)}\}^\infty_{k=1}$ in `ProST-T`. Also, we set o_k to consist of two variables $o_k = (o^p_{(k)}, o^r_{(k)})$. We define a function $o^p_{(k)} \coloneqq \{o : \mathcal{S}\times\mathcal{S}\times\mathcal{A} \to \mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|} \mid o(s',s,a) = P^{(k)}(s'|s,a),\ \forall (s',s,a)\}$, a function $o^r_{(k)} \coloneqq \{o : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^{|\mathcal{S}||\mathcal{A}|} \mid o(s,a) = R^{(k)}(s,a),\ \forall (s,a)\}$. Then, the transition probability and reward value $P^{(k)}, R^{(k)}$, can be constructed by the inner product

of the stationary function φ and the unknown non-stationary function $o^p_{(k)}$ and $\psi, o^r_{(k)}$ as follows,

$$P^{(k)}(s' \mid s,a) \coloneqq < \psi(s',s,a), o^p_{(k)}(s',s,a) > \quad ,\forall (s',s,a) \tag{D.1}$$

$$R^{(k)}(s,a) \coloneqq < \varphi(s,a), o^r_{(k)}(s,a) > \quad ,\forall (s,a) \tag{D.2}$$

For notation simplicity, we denote $< \phi, o^p_{(k)} >, < \varphi, o^r_{(k)} >$ as the inner product of two functions $\phi, o^p_{(k)}$ and $\varphi, o^r_{(k)}$ respectively. Then $P^{(k)} =< \phi, o^p_{(k)} >,\ R^{(k)} =< \varphi, o^r_{(k)} >$ holds.

For intuitive explanation, $o^p_{(k)}$ contains all transition probabilities for all (s',s,a) as a vector form with size $\mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|}$ and $o^r_{(k)}$ contains all rewards for all (s,a) as a vector form with size $\mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$.

2. The agent estimates $o^r_{(k)}, o^p_{(k)}$ rather than observe them.

We have defined the function $o^p_{(k)}, o^r_{(k)}$ as the transition probability and reward function at the episode k respectively. Now the agent tries to estimate $o^p_{(k)}, o^r_{(k)}$ as $\widehat{o}^p_{(k)}, \widehat{o}^r_{(k)}$ from the current trajectory τ_k as follows,

$$\widehat{o}^p_{(k)}(s',s,a) = \frac{n_k(s',s,a)}{\lambda + n_k(s,a)}, \quad \forall (s',s,a) \in \mathcal{S}\times\mathcal{S}\times\mathcal{A}$$

$$\widehat{o}^r_{(k)}(s,a) = \frac{\sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s^k_h, a^k_h)\right] \cdot R^{(k)}_h}{n_k(s,a)}, \quad \forall (s,a) \in \mathcal{S}\times\mathcal{A}$$

where $n_k(s,a)$ $(n_k(s',s,a))$ denotes visitation count of state s and take action a (and move to next state s') through trajectory τ_k. We denote $\widehat{o}^p_{k,h} = \widehat{o}^p_{(k)}(s^k_{h+1}, s^k_h, a^k_h)$ and $\widehat{o}^r_{k,h} = \widehat{r}^k_h(s^k_h, a^k_h)$

Then, it is easily to check that at the episode k, the following holds for state,action pairs from the k^{th} trajectory $\{s^k_0, a^k_0, s^k_1, a^k_1, .., s^k_{H-1}, a^k_{H-1}, s^k_H\}$from equations (D.1), (D.2),

$$P^{(k)}(s^k_{h+1} \mid s^k_h, a^k_h) \coloneqq < \psi(s^k_{h+1}, s^k_h, a^k_h), o^p_{(k)}(s^k_{h+1}, s^k_h, a^k_h) > \quad ,\forall h \in [H] \tag{D.3}$$

$$R^{(k)}(s^k_h, a^k_h) \coloneqq < \varphi(s^k_h, a^k_h), o^r_{(k)}(s^k_h, a^k_h) > \quad ,\forall h \in [H] \tag{D.4}$$

For an intuitive explanation, the observed non-stationary variable $\hat{o}^p_{(k)}, \hat{o}^r_{(k)}$ can be interpreted as the partially observed vector, namely, the agent only estimates some entries of $o^p_{(k)}, o^r_{(k)}$ that $(s',s,a) = (s^k_{h+1}, s^k_h, a^k_h),\ \forall h \in [H]$, and otherwise the agent does not know.

D.1.2 Meta-function f, g

The function f, g estimates and predicts as follows.

1. **Function f**: f forecasts one-episode ahead non-stationary variable $\hat{o}^p_{(k+1)}, \hat{o}^r_{(k+1)}$ by minimizing the following loss function $\mathcal{L}_{f^\diamond}$ with regularization parameter $\lambda \in \mathbb{R}_+$,

$$\mathcal{L}_{f^\diamond}(\phi\ ;\ \widehat{o}^\diamond_{(k-w+1:k)}) = \lambda||\phi||^2 + \sum_{s=k-w+1}^{k} \sum_{h=0}^{H-1} \left((\square^s_h)^\top \phi - \widehat{o}^\diamond_{s,h}\right)$$

   where $\diamond = r, p$ and $\square = \varphi$ if $\diamond = r$ and $\square = \psi$ if $\diamond = p$. We set $\phi^k_{f^\diamond} = \text{argmin}_\phi \mathcal{L}_{f^\diamond}(\widehat{o}^\diamond_{k-(w-1):k})$. We use $\phi^k_{f^\diamond}$ as $\widehat{o}^\diamond_{k+1}$.

2. **Function g**: Then g predicts the function $\widehat{P}^{(k+1)}, \widehat{R}^{(k+1)}$ by the function $\widehat{g}^P_{(k+1)}, \widehat{g}^R_{(k+1)}$ as $\widehat{P}^{(k+1)} = \widehat{g}^P_{(k+1)} \coloneqq < \varphi, \widehat{o}^p_{(k+1)} >$, and $\widehat{R}^{(k+1)} = \widehat{g}^R_{(k+1)} \coloneqq < \varphi, \widehat{o}^r_{k+1} > +2\Gamma^k_w$ where $\Gamma^k_w(s,a) : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ is the exploration bonus term that adapts the counter-based bonus terms in the literature.

We elaborate on above two procedures as follows.

1. The function f solves the optimization problem to provide the future $\widehat{o}_{(k+1)}$.

The meta function $\mathcal{I}_{g\circ f}$ forecasts the $k{+}1^{th}$ episode's non-stationary variable as $(\widehat{o}^p_{(k+1)}, \widehat{o}^r_{(k+1)})$ from $\widehat{o}_{(k-w+1:k)}$ where w is the sliding window length (past reference length). The function f forecasts $o^p_{(k+1)}$, $o^r_{(k+1)}$ by minimizing the following two regularized least squares optimization problems in sliding windows [18].

$$\widehat{o}^p_{(k+1)} = \underset{o\in\mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|}}{\arg\min}\left(\lambda\|o\|^2 + \sum_{s=k-w+1,h=0}^{k,H}\left((\psi^s_h)^\top o - \widehat{o}^p_{s,h}\right)\right) \tag{D.5}$$

$$\widehat{o}^r_{(k+1)} = \underset{o\in\mathbb{R}^{|\mathcal{S}||\mathcal{A}|}}{\arg\min}\left(\lambda\|o\|^2 + \sum_{s=k-w+1,h=0}^{k,H-1}\left((\varphi^s_h)^\top o - \widehat{o}^r_{s,h}\right)\right) \tag{D.6}$$

2. Function g constructs $\widehat{P}^{(k+1)}, \widehat{R}^{(k+1)}$ from $\widehat{o}_{k+1}$.

From the equations (17a) and (17b) of the paper [32], the explicit solutions of (D.5) and (D.6) are represented as follows.

$$\widehat{o}^p_{(k+1)}(s',s,a) = \frac{\sum_{t=k-w+1}^{k} n_t(s',s,a)}{\lambda + \sum_{t=k-w+1}^{k} n_t(s,a)},\ \widehat{o}^r_{(k+1)}(s,a) = \frac{\sum_{t=k-w+1}^{k}\sum_{h=0}^{H-1}\mathbb{1}\left[(s,a)=(s^t_h,a^t_h)\right]\cdot r^t_h}{\lambda + \sum_{t=k-w+1}^{k} n_t(s,a)} \tag{D.7}$$

Then, the `ProST-T` predicts the future model using the function $\widehat{g}^P_{k+1}, \widehat{g}^R_{k+1}$ as follows,

$$\begin{aligned}
\widehat{g}^P_{k+1}(s',s,a) &:= <\varphi(s',s,a), \widehat{o}^p_{(k+1)}(s',s,a)> \\
\widetilde{g}^R_{k+1}(s,a) &:= <\varphi(s,a), \widehat{o}^r_{(k+1)}(s,a)> \\
\widehat{g}^R_{k+1}(s,a) &:= \widetilde{g}^R_{k+1}(s,a) + 2\Gamma^k_w(s,a)
\end{aligned}$$

Recall that the exploration bonus term $\Gamma^k_w(s,a) : \mathcal{S}\times\mathcal{A}\to\mathbb{R}$ is choosed $\Gamma^k_w(s,a) = \beta\left(\sum_{t=k-w+1}^{k} n_t(s,a)+\lambda\right)^{-1/2}$, $\beta > 0$. Then, the function g. the future MDP's $\widehat{P}^{(k+1)}, \widehat{R}^{(k+1)}$ by $\widehat{g}^P_{k+1}, \widehat{g}^R_{k+1}$, respectively. To understand the following theoretic analysis much better, we let the notation $\widehat{P}^{(k+1)} = \widehat{g}^P_{k+1}$, $\widetilde{R}^{(k+1)} = \widetilde{g}^R_{k+1}$, $\widehat{R}^{(k+1)} = \widehat{g}^R_{k+1}$.

D.1.3 Baseline algorithm A

The `ProST-T` utilizes softmax parameterization that naturally ensures that the policy lies in the probability simplex. For any function that satisfies $\theta : \mathcal{S}\times\mathcal{A}\to\mathbb{R}$, the policy π^k is generated by the softmax transformation of θ_k at episode k. Furthermore, to promote exploration and discourage premature convergence to suboptimal policies in a non-stationary environment, we implemented a widely used strategy known as entropy regularization. We augment the future state value function with an additional $\pi^k(s)$ entropy term, denoted by $\tau\mathcal{H}(s,\pi^k)$, where $\tau > 0$. We have performed a theoretical analysis with two baseline algorithms : Natural Policy Gradient (NPG) `Alg` and Natural Policy Gradient (NPG) with entropy regularization $\texttt{Alg}_\tau$

Softmax Parameterization. For any function that satisfies $\theta : \mathcal{S}\times\mathcal{A}\to\mathbb{R}$, the policy π^k is generated by the softmax transformation of θ_k at episode k. We let the notation $\pi^k = \pi_{\theta^k}$ and soft parameterizeation is defined as follows :

$$\pi_{\theta^k}(a|s) := \frac{\exp\left(\theta^k(s,a)\right)}{\sum_{a'\in\mathcal{A}}\exp\left(\theta^k(s,a')\right)}\quad \forall(s,a)\in\mathcal{S}\times\mathcal{A}$$

Under softmax parameterization, the NPG update rule admits a simple form of update rule as line 17 of Algorihtm 2 in Appendix F.1. This is elaborated in [21]. **Entropy regularized value maximization.** We define *forecasted* entropy-regularized state value function $\widehat{V}^{\pi,k+1}_\tau(s)$ as

$$\widehat{V}^{\pi,k+1}_\tau(s) := \widehat{V}^{\pi,k+1}(s) + \tau\mathcal{H}(s,\pi)$$

where $\tau\geq 0$ denotes the regularization parameter and $\mathcal{H}(s,\pi)$ is a discounted entropy as follows :

$$\mathcal{H}(s,\pi) := \mathbb{E}_{\widehat{\mathcal{M}}_{(k+1)}}\left[\sum_{h=0}^{H}-\gamma^h\log\pi(\hat{a}_h|\hat{s}_h)|\hat{s}_0 = s\right]$$

Also, we define the *forecasted* regularized Q-function $\widehat{Q}_\tau^{\pi,k+1}$ as

$$\widehat{Q}_\tau^{\pi,k+1}(s,a) = \hat{r}_h^{k+1} + \gamma \mathbb{E}_{s' \sim \widehat{P}^{k+k_f}(\cdot|s,a)} \left[\widehat{V}_\tau^{\pi,k+1}(s') \right]$$
$$\text{where } (s', s, a) = (\hat{s}_{h+1}^{k+k_f}, \hat{s}_h^{k+1}, \hat{a}_h^{k+1})$$

D.2 Notation for theoretical analysis.

This subsection introduces some notations that we only use in the proofs.

Let us define the *forecasting model error* $\Delta_k^r(s,a)$ and *forecasting transition probability model error* $\Delta_k^p(s,a)$ as follow. Recall that $\widetilde{R}(s,a)$, $\widehat{P}^{(k+1)}$ are the estimation of future reward and transition probability by solving the optimization problem (D.5), (D.6)

$$\Delta_k^r(s,a) \coloneqq \left| \left(R^{(k+1)} - \widetilde{R}^{(k+1)} \right)(s,a) \right| \tag{D.8}$$
$$\Delta_k^p(s,a) \coloneqq \left\| \left(P^{(k+1)} - \widehat{P}^{(k+1)} \right)(\cdot \mid s,a) \right\|_1 \tag{D.9}$$

and define the model error that considers the bonus term as follows,

$$\Delta_k^{Bonus,r}(s,a) \coloneqq \left| \left(R^{(k+1)} - \widehat{R}^{(k+1)} \right)(s,a) \right|$$

where we have defined $\widehat{R}(s,a) = \widetilde{R}(s,a) + 2\Gamma_w^k(s,a)$. Recall the definition of *empirical* forecasting reward model error $\bar{\Delta}_{k,h}^r$, *empirical* forecasting transition probability model error $\bar{\Delta}_{k,h}^p$,

$$\bar{\Delta}_{k,h}^r \coloneqq \left| \left(R^{(k+1)} - \widetilde{R}^{(k+1)} \right)(s_h^{k+1}, a_h^{k+1}) \right|$$
$$\bar{\Delta}_{k,h}^p \coloneqq \left\| \left(P^{(k+1)} - \widehat{P}^{(k+1)} \right)(\cdot \mid s_h^{k+1}, a_h^{k+1}) \right\|_1$$

and the empirical bonus considered reward model error

$$\bar{\Delta}_{k,h}^{Bonus,r} \coloneqq \left| \left(R^{(k+1)} - \widehat{R}^{(k+1)} \right)(s_h^{k+1}, a_h^{k+1}) \right|$$

and the *total empirical* forecasting reward model error $\bar{\Delta}_K^r$, *total empirical* forecasting transition probability model error $\bar{\Delta}_K^p$ as follows.

$$\bar{\Delta}_K^r \coloneqq \sum_{k=1}^{K-1} \sum_{h=0}^{H-1} \bar{\Delta}_{k,h}^r \tag{D.10}$$
$$\bar{\Delta}_K^p \coloneqq \sum_{k=1}^{K-1} \sum_{h=0}^{H-1} \bar{\Delta}_{k,h}^p \tag{D.11}$$

We also define the variable $\Lambda_w^k(s,a)$ that quantifies the visitation as follows.

$$\Lambda_w^k(s,a) = \left[\lambda + \sum_{t=(1 \wedge k-w+1)}^{k} n_t(s,a) \right]^{-1} \tag{D.12}$$

Then, it is easily to check that

$$\Gamma_w^k(s,a) = \beta \sqrt{\Lambda_w^k(s,a)} \tag{D.13}$$

holds.

We define $r_{\max}, \widetilde{r}_{\max}, R_{\max}^{(k+1)}, \widetilde{R}_{\max}^{(k+1)}$ as follows,

$$R_{\max}^{(k+1)} \coloneqq \|R^{(k+1)}(s,a)\|_\infty, \; r_{\max} \coloneqq \max_{1 \le k \le K-1} (R_{\max}^{(k+1)}),$$
$$\widetilde{R}_{\max}^{(k+1)} \coloneqq \|\widetilde{R}^{(k+1)}(s,a)\|_\infty, \; \widetilde{r}_{\max} \coloneqq \max_{1 \le k \le K-1} \widetilde{R}_{\max}^{(k+1)}$$

and since $\|\widehat{R}^{(k+1)}(s,a)\|_\infty \le \|\widetilde{R}^{(k+1)}(s,a)\|_\infty + \|2\Gamma_w^k(s,a)\|_\infty = \widetilde{R}_{\max}^{(k+1)} + \frac{2\beta}{\sqrt{\lambda}}$ holds, we define $\hat{r}_{\max}^{k+1}$ as follows.

$$\hat{r}_{\max}^{k+1} \coloneqq \widetilde{R}_{\max}^{(k+1)} + \frac{2\beta}{\sqrt{\lambda}}$$

Also, since β, λ are hyperparameters independent of k, the following holds.

$$\hat{r}_{\max} \coloneqq \widetilde{r}_{\max} + \frac{2\beta}{\sqrt{\lambda}} \tag{D.14}$$

D.3 Proofs

Proof of Theorem 1. Following the definition of the dynamic regret (Definition C.1),it can be separated into three terms.

$$
\begin{aligned}
&\mathfrak{R}\left(\{\widehat{\pi}^{k+1}\}_{1:K-1}, K\right)\\
&\coloneqq \sum_{k=1}^{K-1}\left(V^{*,k+1}(s_0) - V^{\widehat{\pi}^{k+1},k+1}(s_0)\right)\\
&= \underbrace{\sum_{k=1}^{K-1}\left(V^{*,k+1}(s_0) - \widehat{V}^{*,k+1}(s_0)\right)}_{①} + \underbrace{\sum_{k=1}^{K-1}\left(\widehat{V}^{*,k+1}(s_0) - \widehat{V}^{\widehat{\pi}^{k+1},k+1}(s_0)\right)}_{②}\\
&+ \underbrace{\sum_{k=1}^{K-1}\left(\widehat{V}^{\widehat{\pi}^{k+1},k+1}(s_0) - V^{\widehat{\pi}^{k+1},k+1}(s_0)\right)}_{③}
\end{aligned}
$$

1. Upper bound of ①. The gap between $V^{\pi^{*,k+1},k+1}(s_0)$ and $\widehat{V}^{\widehat{\pi}^{*,k+1},k+1}(s_0)$ comes from the gap of two optimal value functions evaluated from two different MDPs : $\mathcal{M}_{(k+1)}$, $\widehat{\mathcal{M}}_{(k+1)}$.

We first come up with the upper bound of the difference between $Q_h^{*,k+1}(s,a)$ and $\widehat{Q}_h^{*,k+1}(s,a)$ for any $(s,a) \in \mathcal{S}\times\mathcal{A}$. The difference can be separated into three terms as follows:

$$
\begin{aligned}
Q_h^{*,k+1}(s,a) - \widehat{Q}_h^{*,k+1}(s,a) \le &\underbrace{\|Q_h^{*,k+1}(s,a) - Q_\infty^{*,k+1}(s,a)\|_\infty}_{(1.1)} + \underbrace{\left(Q_\infty^{*,k+1}(s,a) - \widehat{Q}_\infty^{*,k+1}(s,a)\right)}_{(1.2)}\\
&+ \underbrace{\|\widehat{Q}_h^{*,k+1}(s,a) - \widehat{Q}_\infty^{*,k+1}(s,a)\|_\infty}_{(1.3)}
\end{aligned}
$$

1.1.Term (1.1) and (1.3).

First, the term (1.1) and (1.3) can be bounded as follows,

$$
\begin{aligned}
(1.1) &= \left\|\mathbb{E}_{\mathcal{M}_{(k+1)},\pi^*}\Big[\sum_{i=0}^{H-h-1}\gamma^i r_{i+h}^{k+1} - \sum_{i=0}^{\infty}\gamma^i r_i^{k+1} \mid s_h^{k+1} = s, a_h^{k+1} = a\Big]\right\|_\infty\\
&\le \left|\sum_{i=H-h}^{\infty}\gamma^i r_{\max}\right|\\
&= \frac{\gamma^{H-h}}{1-\gamma} r_{\max}
\end{aligned}
$$

Through the similar process, we can also obtain the upper bound (1.3) $\le \gamma^{H-h}/(1-\gamma)\hat{r}_{\max}$.

1.2. Term (1.2).

The upper bound of the term (1.2) can be bounded utilizing $\bar{\iota}_\infty^{k+1}(s,a)$ (Def (C.2)). Then, the Q function gap between $Q_\infty^{*,k}$, $\widehat{Q}_\infty^{*,k}$ can be represented using the bellman equation as follows,

$$
\begin{aligned}
(1.2) &= \left(Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1}\right)(s,a) && \text{(D.15)}\\
&= \left(R^{(k+1)} + \gamma P^{(k+1)} V_\infty^{*,k+1}\right)(s,a) - \widehat{Q}_\infty^{*,k+1}(s,a) && \text{(D.16)}\\
&= \left(R^{(k+1)} + \gamma P^{(k+1)} \widehat{V}_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1}\right)(s,a) + \gamma P^{(k+1)}\left(V_\infty^{*,k+1} - \widehat{V}_\infty^{*,k+1}\right)(s,a)\\
&\le \bar{\iota}_\infty^{k+1}(s,a) + \gamma P^{(k+1)}\left(V_\infty^{*,k+1} - \widehat{V}_\infty^{*,k+1}\right)(s,a)\\
&= \bar{\iota}_h^{k+1}(s,a) + \gamma P^{(k+1)}\left(\langle Q_\infty^{*,k+1}, \pi^{*,k+1}\rangle_{\mathcal{A}} - \langle \widehat{Q}_\infty^{*,k+1}, \widehat{\pi}^{*,k+1}\rangle_{\mathcal{A}}\right)(s,a) && \text{(D.17)}\\
&= \bar{\iota}_\infty^{k+1}(s,a) + \gamma P^{(k+1)}\left(\langle Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1}, \pi^{*,k+1}\rangle_{\mathcal{A}} + \langle \widehat{Q}_\infty^{*,k+1}, \pi^{*,k+1} - \widehat{\pi}^{*,k+1}\rangle_{\mathcal{A}}\right)(s,a)\\
&\le \bar{\iota}_\infty^{k+1}(s,a) + \gamma P^{(k+1)}\left(\langle Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1}, \pi^{*,k+1}\rangle_{\mathcal{A}}\right)(s,a) && \text{(D.18)}
\end{aligned}
$$

where equation (D.16) and (D.17) holds by the definition of bellman equations (equation (C.5) and (C.9)). Equation (D.18) holds by $\langle \widehat{Q}_\infty^{*,k+1}, \pi^{*,k+1} - \widehat{\pi}^{*,k+1}\rangle_{\mathcal{A}}(s,a) \le 0$ since $\widehat{\pi}^{*,k+1}$ is the optimal policy of $\widehat{Q}_\infty^{*,k+1}$. We now define the matrix operator $(\mathbb{P}\circ\pi)(s,a) : \mathbb{R}^{|\mathcal{S}||\mathcal{A}|} \to \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ as the transition matrix that state action pair transition from (s,a) to (s',a') when following the policy π in an environment with transition probability $\mathbb{P}$. Also, define the one-vector $\mathbb{1}_{(s,a)} \in \mathbb{R}^{|\mathcal{S}|\mathcal{A}|}$ that the (s,a) entity is one and otherwise zero. Then the equation (D.15) is the same as the $(s,a)^{\text{th}}$ entity of the vector $\mathbb{1}_{(s,a)} \cdot (Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1})(s,a)$. Also, the RHS of equation (D.18) can be represented as

$$
\begin{aligned}
P^{(k+1)}\left(\langle Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1}, \pi^{*,k+1}\rangle_{\mathcal{A}}\right)(s,a) &= \left(P^{(k+1)} \circ \pi^{*,k+1}\right)\left(\mathbb{1}_{(s,a)} \cdot (Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1})\right)(s,a)\\
&= \left(\mathbb{P}_{\pi^*}^{k+1}\right)\left(\mathbb{1}_{(s,a)} \cdot (Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1})\right)(s,a)
\end{aligned}
$$

where we denote $P^{(k+1)} \circ \pi^{*,k+1} \coloneqq \mathbb{P}_{\pi^*}^{k+1}$ for notation simplicity.

Then, we can reformulate the inequality (between equation (D.15) and (D.18)) into a vector form where inequality holds for element-wise over $\forall(s,a)$:

$$
\left(\mathbb{1}_{(s,a)} \cdot \left(Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1}\right)\right)(s,a) \le \mathbb{1}_{(s,a)} \cdot \bar{\iota}_\infty^{k+1}(s,a) + \gamma(\mathbb{P}_{\pi^*}^{k+1})\left(\mathbb{1}_{(s,a)} \cdot (Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1})\right)(s,a)
$$

Then, rearranging the above inequality yields the following.

$$
\begin{aligned}
\mathbb{1}_{(s,a)} \cdot \left(Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1}\right)(s,a) &\le (\mathbb{I} - \gamma\mathbb{P}_{\pi^*}^{k+1})^{-1}\mathbb{1}_{(s,a)} \cdot \bar{\iota}_\infty^{k+1}(s,a) && \text{(D.19)}\\
&= \frac{1}{1-\gamma}\bar{\iota}_\infty^{k+1}(s,a)
\end{aligned}
$$

Now, note that $(\mathbb{I} - \gamma\mathbb{P}_{\pi^*}^{k+1})^{-1}$ can be expanded with infinite summation of matrix operator $P^{(k+1)} \circ \pi^{*,k+1}$ as $(\mathbb{I} - \gamma\mathbb{P}_{\pi^*}^{k+1})^{-1} = \mathbb{I} + \gamma\mathbb{P}_{\pi^*}^{k+1} + (\gamma\mathbb{P}_{\pi^*}^{k+1})^2 + \ldots$. Also, $\mathbb{1}_{(s,a)}$ can be viewed as the Dirac delta state action distribution that always yields (s,a), then $\nu_{(s,a)}^{\pi^{*,k+1},k+1} = (\mathbb{I} - \gamma\mathbb{P}_{\pi^*}^{k+1})^{-1}\mathbb{1}_{(s,a)}$ holds, where ν is the unnormalized occupancy measure of (s,a) from the definition (C.10). Then taking the l_1 norm over the inequality (D.19) yields the following.

$$
\begin{aligned}
\|\mathbb{1}_{(s,a)} \cdot \left(Q_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1}\right)(s,a)\|_1 &\le \|(\mathbb{I} - \gamma\mathbb{P}_{\pi^*}^{k+1})^{-1}\mathbb{1}_{(s,a)} \cdot \bar{\iota}_\infty^{k+1}(s,a)\|_1\\
&= \|(\mathbb{I} - \gamma\mathbb{P}_{\pi^*}^{k+1})^{-1}\mathbb{1}_{(s,a)}\|_1 \cdot |\bar{\iota}_\infty^{k+1}(s,a)|\\
&= \frac{1}{1-\gamma}|\bar{\iota}_\infty^{k+1}(s,a)| && \text{(D.20)}
\end{aligned}
$$

Equation (D.20) holds since $\nu_{(s,a)}^{\pi^{*,k+1},k+1}$ is unnormalized probability distribution.

Then, for any $(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$, we obtain the following inequality by combining the term (1.1), (1.2), (1.3).

$$
Q_h^{*,k+1}(s,a) - \widehat{Q}_h^{*,k+1}(s,a) \le \frac{\gamma^{H-h}}{1-\gamma}(r_{\max} + \hat{r}_{\max}) + \frac{1}{1-\gamma}|\bar{\iota}_\infty^{k+1}(s,a)|
$$

1.3. Combining the terms (1.1), (1.2), (1.3).

Finally, the upper bound of ① is given as follows,

$$
\begin{aligned}
① &= \sum_{k=1}^{K-1} \left(V^{\pi^{*,k+1},k+1}(s_0) - \widehat{V}^{\widehat{\pi}^{*,k+1},k+1}(s_0) \right) \\
&\leq \sum_{k=1}^{K-1} \left\| Q^{*,k+1} - \widehat{Q}^{*,k+1} \right\|_\infty \\
&= \sum_{k=1}^{K-1} \cdot \frac{\gamma^H}{1-\gamma}(r_{\max} + \hat{r}_{\max}) + \frac{1}{1-\gamma} \sum_{k=1}^{K-1} \|\bar{\iota}_\infty^{k+1}\|_\infty \\
&= (K-1) \cdot \frac{\gamma^H}{1-\gamma}(r_{\max} + \hat{r}_{\max}) + \frac{1}{1-\gamma} \bar{\iota}_\infty^K
\end{aligned} \tag{D.21}
$$

where we have defined $\bar{\iota}_\infty^K \coloneqq \sum_{k=1}^{K-1} \|\bar{\iota}_\infty^{k+1}\|_\infty$ in the theorem 1.

2. Upper bound of ②.

The gap between $\widehat{V}^{*,k+1}(s_0)$ and $\widehat{V}^{\widehat{\pi}^{k+1},k+1}(s_0)$ comes from optimization error between a optimal policy $\widehat{\pi}^{*,k+1}$ and a policy $\widehat{\pi}^{k+1}$ which are both driven from a same MDP $\widehat{\mathcal{M}}_{(k+1)}$. We also separate its gap into following three terms,

$$
\begin{aligned}
\text{②'s } (k)^{th} \text{ term} &= \widehat{V}^{*,k+1}(s_0) - \widehat{V}^{\widehat{\pi}^{k+1},k+1}(s_0) \\
&= \left(\widehat{V}^{*,k+1}(s_0) - \widehat{V}_\infty^{*,k+1}(s_0) \right) + \left(\widehat{V}_\infty^{*,k+1}(s_0) - \widehat{V}_\infty^{\widehat{\pi}^{k+1},k+1}(s_0) \right) + \\
&+ \left(\widehat{V}_\infty^{\widehat{\pi}^{k+1},k+1}(s_0) - \widehat{V}^{\widehat{\pi}^{k+1},k+1}(s_0) \right) \qquad \text{(D.22)} \\
&\leq \underbrace{\left(\widehat{V}_\infty^{*,k+1}(s_0) - \widehat{V}_\infty^{\widehat{\pi}^{k+1},k+1}(s_0) \right)}_{②.1} + \frac{2\gamma^H \widehat{r}_{\max}}{1-\gamma} \qquad \text{(D.23)}
\end{aligned}
$$

where we denote underscprit ∞ in the notation $\widehat{V}_\infty^{\pi,k+1}(s_0), \widehat{V}_{\infty,\tau}^{\pi,k+1}(s_0)$ to be forecasted value function, forecasted entropy-regularized value function when $H = \infty$ (infinite horizon MDPs). Equation (D.22) holds since $\widehat{V}^{\pi,k+1}(s) - \widehat{V}_\infty^{\pi,k+1}(s) = \mathbb{E}_{\widehat{\mathcal{M}}_{(k+1)},\pi}\left[\sum_{h=H}^\infty \gamma^h \widehat{r}_h^{k+1} \;\middle|\; s = \widehat{s}_0^{k+1} \right] \leq \frac{\gamma^H}{1-\gamma} \widehat{r}_{\max}$ holds for any $\pi \in \Pi$.

2.1. Upper bound ② - NPG without entropy regularization (`Alg`). The term ②.1 in the equation (D.23) can be bounded as follows,

$$
\begin{aligned}
②.1 &= \widehat{V}_\infty^{*,k+1}(s_0) - \widehat{V}_\infty^{\widehat{\pi}^{k+1},k+1}(s_0) \\
&\leq \frac{\log|\mathcal{A}|}{\eta G} + \frac{1}{(1-\gamma)^2 G}
\end{aligned} \tag{D.24}
$$

equation (D.24) holds from Theorem (5.3) of the paper [39]. Now, combining the equation (D.23,D.24) offers the upper bound of the term ②'s $k^{(th)}$ term as follows,

$$
\begin{aligned}
\text{②'s } (k)^{th} \text{ term} &= \widehat{V}^{\widehat{\pi}^{*,k+1},k+1}(s_0) - \widehat{V}^{\widehat{\pi}^{k+1},k+1}(s_0) \\
&\leq \frac{1}{(1-\gamma)^2 G} + \frac{\log|\mathcal{A}|}{\eta G} + \frac{2\gamma^H \widehat{r}_{\max}}{1-\gamma}
\end{aligned}
$$

and the upper bound of the term ②.

$$
\begin{aligned}
② &= \sum_{k=1}^{K-1} \left(\widehat{V}^{\widehat{\pi}^{*,k+1},k+1}(s_0) - \widehat{V}^{\widehat{\pi}^{k+1},k+1}(s_0) \right) \\
&\leq (K-1) \left(\frac{1}{(1-\gamma)^2 G} + \frac{\log|\mathcal{A}|}{\eta G} + \frac{2\gamma^H \widehat{r}_{\max}}{1-\gamma} \right)
\end{aligned} \tag{D.25}
$$

2.2. Upper bound ② - NPG with entropy regularization ($\texttt{Alg}_\tau$).

Term (2.1) in the equation (D.23) can be further bounded as follows,

$$
\begin{aligned}
(2.1) &= \widehat{V}_{\infty}^{*,k+1}(s_0) - \widehat{V}_{\infty}^{\widehat{\pi}^{k+1},k+1}(s_0) \\
&= \left(\widehat{V}_{\infty}^{*,k+1}(s_0) - \widehat{V}_{\infty,\tau}^{*,k+1}(s_0)\right) + \left(\widehat{V}_{\infty,\tau}^{*,k+1}(s_0) - \widehat{V}_{\infty,\tau}^{\widehat{\pi}^{k+1},k+1}(s_0)\right) \\
&+ \left(\widehat{V}_{\infty,\tau}^{\widehat{\pi}^{k+1},k+1}(s_0) - \widehat{V}_{\infty}^{\widehat{\pi}^{k+1},k+1}(s_0)\right) \\
&\leq \left\|\widehat{V}_{\infty}^{*,k+1}(s_0) - \widehat{V}_{\infty,\tau}^{*,k+1}(s_0)\right\|_{\infty} + \left\|\widehat{V}_{\infty,\tau}^{*,k+1}(s_0) - \widehat{V}_{\infty,\tau}^{\widehat{\pi}^{k+1},k+1}(s_0)\right\|_{\infty} \\
&+ \left\|\widehat{V}_{\infty,\tau}^{\widehat{\pi}^{k+1},k+1}(s_0) - \widehat{V}_{\infty}^{\widehat{\pi}^{k+1},k+1}(s_0)\right\|_{\infty} \\
&\leq \underbrace{\left\|\widehat{V}_{\infty,\tau}^{*,k+1}(s_0) - \widehat{V}_{\infty,\tau}^{\widehat{\pi}^{k+1},k+1}(s_0)\right\|_{\infty}}_{(2.2)} + \frac{2\tau\log|\mathcal{A}|}{1-\gamma}
\end{aligned} \tag{D.26}
$$

where equation (D.26) holds since $\left\|\widehat{V}_{\infty}^{\pi,k+1}(s_0) - \widehat{V}_{\infty,\tau}^{\pi,k+1}(s_0)\right\|_{\infty} = \tau\max_s|\mathcal{H}(s,\pi)| \leq \tau\frac{\log|\mathcal{A}|}{1-\gamma}$ holds for $\forall\pi$.

Now, let's come up with a upper bound of the term (2.2) of equation (D.26). Now, following the policy-update rule of `ProST-T` (Algorithm 2 in Appendix F.2), suppose for given $g \in [G]$, we got *inexact* soft Q-function value of the policy $\widehat{\pi}^{(\Delta_\pi)}$ as $\widetilde{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}$ where $\widehat{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}$ denotes *exact* soft Q-function value. The approximation gap $|\widetilde{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}} - \widehat{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}|$ can come from computing Q using finite samples. For choosen hyperparameter δ, Let the maximum of approximation gap over (s,a) is smaller than δ, namely $||\widetilde{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}} - \widehat{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}||_{\infty} \leq \delta$ holds. Then, the policy-update rule of `ProST-T` can be written as follows :

$$
\begin{aligned}
&\widehat{\pi}^{(g+1)}(\cdot|s) = \frac{1}{Z^{(\Delta_\pi)}}\cdot\left(\widehat{\pi}^{(\Delta_\pi)}(\cdot|s)\right)^{1-\frac{\eta\tau}{1-\gamma}}\exp\left(\frac{\eta\widetilde{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}(s,a)}{1-\gamma}\right) \\
&\text{where } \ ||\widetilde{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}(s,a) - \widehat{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}(s,a)||_{\infty} \leq \delta \ \text{ for } \ \forall(s,a) \in \mathcal{S}\times\mathcal{A}
\end{aligned}
$$

where $Z^{(\Delta_\pi)}(s) = \sum_{a\in\mathcal{A}}\left(\widehat{\pi}^{(\Delta_\pi)}(a|s)\right)^{1-\frac{\eta\tau}{1-\gamma}}\exp\left((\eta\widehat{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}(s,a))/(1-\gamma)\right)$

From the Theorem 2 of the paper [21], when the learning rate satisfies $0 \leq \eta \leq (1-\gamma)/\tau$, then the approximate entropy-regularized NPG method satisfies the linear convergence theorem as follows:

$$
||\widehat{Q}_{\tau}^{*,k+1} - \widehat{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}||_{\infty} \leq \gamma\left[(1-\eta\tau)^{\Delta_\pi - 1}C_1 + C_2\right] \tag{D.27}
$$

$$
||\log\widehat{\pi}^{*,k+1} - \log\widehat{\pi}^{(\Delta_\pi)}||_{\infty} \leq 2\tau^{-1}\left[(1-\eta\tau)^{\Delta_\pi - 1}C_1 + C_2\right] \tag{D.28}
$$

where C_1, C_2 is given by

$$
\begin{aligned}
C_1 &= ||\widehat{Q}_{\tau}^{*,k+1} - \widehat{Q}_{\tau}^{\widehat{\pi}^{(0)}}||_{\infty} + 2\tau\left(1 - \frac{\eta\tau}{1-\gamma}\right)||\log\widehat{\pi}^{*,k+1} - \log\widehat{\pi}^{(0)}||_{\infty} \\
&= ||\widehat{Q}_{\tau}^{*,k+1} - \widehat{Q}_{\tau}^{\pi^k}||_{\infty} + 2\tau\left(1 - \frac{\eta\tau}{1-\gamma}\right)||\log\widehat{\pi}^{*,k+1} - \log\widehat{\pi}^{k}||_{\infty}
\end{aligned} \tag{D.29}
$$

$$
C_2 = \frac{2\delta}{1-\gamma}\left(1 + \frac{\gamma}{\eta\tau}\right) \tag{D.30}
$$

The equation (D.29) holds since the policy that the agent executes at the episode k, i.e. π^k or $\widehat{\pi}^{(k-1)+1}$, is same as the initial policy of the policy iteration, i.e. $\widehat{\pi}^{(0)}$ at the episode k. Also, the policy that the agent executes at the episode $k+1$, i.e. $\widehat{\pi}^{k+1}$, is same as the policy after G steps of soft policy iteration, i.e. $\widehat{\pi}^{(\Delta_\pi)}$ at the episode k.

Then, the term (2.2) can be bounded as follows.

$$
\begin{aligned}
\textcircled{2.2} &= \|\widehat{V}_{\tau}^{*,k+1} - \widehat{V}_{\tau}^{\widehat{\pi}^{k+1}}\|_{\infty} \\
&= \|\widehat{V}_{\tau}^{*,k+1} - \widehat{V}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}\|_{\infty} \\
&\leq \|\widehat{Q}_{\tau}^{*,k+1} - \widehat{Q}_{\tau}^{\widehat{\pi}^{(\Delta_\pi)}}\|_{\infty} + \tau\|\log\widehat{\pi}^{*,k+1} - \log\widehat{\pi}^{(\Delta_\pi)}\|_{\infty} \\
&\leq (\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi - 1}C_1 + C_2\right] \qquad \text{(D.31)}
\end{aligned}
$$

Then combining the equation (D.23,D.26,D.31) offers the upper bound of the term ②'s $k^{(th)}$ term as follows,

$$
\begin{aligned}
\text{②'s } (k)^{th} \text{ term} &= \widehat{V}^{\widehat{\pi}^{*,k+1},k+1}(s_0) - \widehat{V}^{\widehat{\pi}^{k+1},k+1}(s_0) \\
&\leq (\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi - 1}C_1 + C_2\right] + \frac{2\gamma^H \widehat{r}_{\max}}{1-\gamma} + \frac{2\tau\log|\mathcal{A}|}{1-\gamma} \qquad \text{(D.32)}
\end{aligned}
$$

and the upper bound of the term ②.

$$
\begin{aligned}
② &= \sum_{k=1}^{K-1}\left(\widehat{V}^{\widehat{\pi}^{*,k+1},k+1}(s_0) - \widehat{V}^{\widehat{\pi}^{k+1},k+1}(s_0)\right) \\
&\leq (K-1)\left((\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi - 1}C_1 + C_2\right] + \frac{2\gamma^H \widehat{r}_{\max}}{1-\gamma} + \frac{2\tau\log|\mathcal{A}|}{1-\gamma}\right) \qquad \text{(D.33)}
\end{aligned}
$$

where the equation (D.32), (D.33) only holds when $0 \leq \eta \leq (1-\gamma)/\tau$

3. Upper bound of ③.

Recall from the Definition (C.3), note that $\iota_h^{k+1}(\widehat{s}_h^{k+1}, \widehat{a}_h^{k+1})$ is a *empirical* estimated model prediction error, the gap between $\mathcal{M}_{(k+1)}$ and $\widehat{\mathcal{M}}_{(k+1)}$. Specifically, at episode k, the meta-algorithm creates the future MDP $\widehat{\mathcal{M}}_{(k+1)}$ and evaluates $\widehat{V}, \widehat{Q}$ using $\widehat{\pi}^{k+1}$, then at episode $k+1$, the agent uses $\widehat{\pi}^{k+1}$ to rollout a trajectory $\{s_0^{k+1}, a_0^{k+1}, s_1^{k+1}, a_1^{k+1}, ..., s_{H-1}^{k+1}, a_{H-1}^{k+1}, s_H^{k+1}\}$. Its term can be expanded as follows,

$$
\begin{aligned}
\iota_h^{k+1}(s_h^{k+1}, a_h^{k+1}) =& R^{(k+1)}(s_h^{k+1}, a_h^{k+1}) + \gamma(P^{(k+1)}\widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1})(s_h^{k+1}, a_h^{k+1}) - \widehat{Q}_h^{\widehat{\pi}^{k+1},k+1}(s_h^{k+1}, a_h^{k+1}) \\
=& R^{(k+1)}(s_h^{k+1}, a_h^{k+1}) + \gamma(P^{(k+1)}\widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1})(s_h^{k+1}, a_h^{k+1}) - Q_h^{\widehat{\pi}^{k+1},k+1}(s_h^{k+1}, a_h^{k+1}) \\
&+ Q_h^{\widehat{\pi}^{k+1},k+1}(s_h^{k+1}, a_h^{k+1}) - \widehat{Q}_h^{\widehat{\pi}^{k+1},k+1}(s_h^{k+1}, a_h^{k+1}) \\
=& \gamma P^{(k+1)}(\widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1} - V_{h+1}^{\widehat{\pi}^{k+1},k+1})(s_h^{k+1}, a_h^{k+1}) \\
&+ Q_h^{\widehat{\pi}^{k+1},k+1}(s_h^{k+1}, a_h^{k+1}) - \widehat{Q}_h^{\widehat{\pi}^{k+1},k+1}(s_h^{k+1}, a_h^{k+1}) \qquad \text{(D.34)}
\end{aligned}
$$

Equation (D.34) holds from the future-bellman equation (C.8) holds for (s,a). Now, define the operator $\widehat{\mathfrak{I}}^{k+1}$ for any function $f : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ as follows.

$$
(\widehat{\mathfrak{I}}^{k+1} f)(s) \coloneqq \langle f(s,\cdot), \widehat{\pi}^{k+1}(\cdot|s)\rangle_{\mathcal{A}}
$$

Recall that $\widehat{V}_h^{\widehat{\pi}^{k+1},k+1}(s) = \left\langle \widehat{Q}_h^{\widehat{\pi}^{k+1},k+1}, \widehat{\pi}^{k+1}\right\rangle_{\mathcal{A}}$ and $V_h^{\widehat{\pi}^{k+1},k+1}(s) = \left\langle Q_h^{\widehat{\pi}^{k+1},k+1}, \widehat{\pi}^{k+1}\right\rangle_{\mathcal{A}}$ holds from equation (C.8) and (C.4) respectively. Then, the gap between $\widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1}(s_h^{k+1})$ and $V_{h+1}^{\widehat{\pi}^{k+1},k+1}(s_h^{k+1})$ can be expanded as follows,

$$
\begin{aligned}
&\widehat{V}_h^{\widehat{\pi}^{k+1},k+1}(s_h^{k+1}) - V_h^{\widehat{\pi}^{k+1},k+1}(s_h^{k+1}) \\
&= \left(\widehat{\mathfrak{I}}^{k+1}\left(\widehat{Q}_h^{\widehat{\pi}^{k+1},k+1} - Q_h^{\widehat{\pi}^{k+1},k+1}\right)\right)(s_h^{k+1}) \\
&= \left(\widehat{\mathfrak{I}}^{k+1}\left(\widehat{Q}_h^{\widehat{\pi}^{k+1},k+1} - Q_h^{\widehat{\pi}^{k+1},k+1}\right)\right)(s_h^{k+1}) - \iota_h^{k+1}(s_h^{k+1}, a_h^{k+1}) \\
&+ \gamma P^{(k+1)}(\widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1} - V_{h+1}^{\widehat{\pi}^{k+1},k+1})(s_h^{k+1}, a_h^{k+1}) + \left(Q_h^{\widehat{\pi}^{k+1},k+1} - \widehat{Q}_h^{\widehat{\pi}^{k+1},k+1}\right)(s_h^{k+1}, a_h^{k+1})
\end{aligned}
$$

Now, we define two sequences $\{D_{h,1}^{k+1}\}, \{D_{h,1}^{k+1}\}$ where $(k,h) = (0,0),(0,1),...,(K-1,H)$. We define $D_{h,1}^{k+1}, D_{h,2}^{k+1}$ as follows.

$$D_{h,1}^{k+1} \coloneqq \gamma^h \left(\widehat{\mathfrak{I}}^{k+1}\left(\widehat{Q}_h^{\widehat{\pi}^{k+1},k+1} - Q_h^{\widehat{\pi}^{k+1},k+1}\right)\right)(s_h^{k+1}) - \gamma^h \left(\widehat{Q}_h^{\widehat{\pi}^{k+1},k+1} - Q_h^{\widehat{\pi}^{k+1},k+1}\right)(s_h^{k+1}, a_h^{k+1})$$
$$D_{h,2}^{k+1} \coloneqq \gamma^{h+1} P^{(k+1)}(\widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1} - V_{h+1}^{\widehat{\pi}^{k+1},k+1})(s_h^{k+1}, a_h^{k+1}) - \gamma^{h+1}\left(\widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1} - V_{h+1}^{\widehat{\pi}^{k+1},k+1}\right)(s_{h+1}^{k+1})$$

Therefore, we have the following recursive formula over h,

$$\begin{aligned}&\gamma^h \left(\widehat{V}_h^{\widehat{\pi}^{k+1},k+1} - V_h^{\widehat{\pi}^{k+1},k+1}\right)(s_h^{k+1}) \\ &= D_{h,1}^{k+1} + D_{h,2}^{k+1} + \gamma^{h+1}\left(\widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1} - V_{h+1}^{\widehat{\pi}^{k+1},k+1}\right)(s_{h+1}^{k+1}) - \gamma^h \iota_h^{k+1}(s_h^{k+1}, a_h^{k+1})\end{aligned}$$

and summation over $h = 0, 1, .., H-1$ yields the following expression,

$$\begin{aligned}&\widehat{V}_0^{\widehat{\pi}^{k+1},k+1}(s_0^{k+1}) - V_0^{\widehat{\pi}^{k+1},k+1}(s_0^{k+1}) \\ &= \sum_{h=0}^{H-1}\left(D_{h,1}^{k+1} + D_{h,2}^{k+1}\right) - \sum_{h=0}^{H-1}\gamma^h \iota_h^{k+1}(s_h^{k+1}, a_h^{k+1})\end{aligned}$$

.Now, for every $(k,h) \in [K] \times [H]$, we define $\mathcal{F}_{h,1}^k$ as a $\sigma-$algebra generated by state-action sequences $\{(s_i^\tau, a_i^\tau)\}_{(\tau,i)\in[k-1]\times[H]} \cup \{(s_i^k, a_i^k)\}_{i\in[h]}$ and define $\mathcal{F}_{h,2}^k$ as an σ-algebra generated by $\{(s_i^\tau, a_i^\tau)\}_{(\tau,i)\in[k-1]\times[H]} \cup \{(s_i^k, a_i^k)\}_{i\in[h]} \cup \{s_{h+1}^k\}$. A filtration $\{\mathcal{F}_{h,m}^k\}_{(k,h,m)\in[K]\times[H]\times[2]}$ is a sequence of σ- algebras in a terms of the time index $t(k,h,m) = 2(k-1)H + 2h + m$ such that $\mathcal{F}_{h,m}^k \subset \mathcal{F}_{h',m'}^{k'}$ for every $t(k,h,m) \le t(k',h',m')$. The estimated $\widehat{V}_h^{\pi,k+1}, \widehat{Q}_h^{\pi,k+1}$ are $\mathcal{F}_{1,1}^{k+1}$ measurable since they are forecasted from past k historical trajectories. Now, since $D_{h,1}^{k+1} \in \mathcal{F}_{h,1}^{k+1}$ and $D_{h,2}^{k+1} \in \mathcal{F}_{h,2}^{k+1}$ holds, $\mathbb{E}[D_{h,1}^{k+1}|\mathcal{F}_{h-1,2}^{k+1}] = 0$ and $\mathbb{E}[D_{h,2}^{k+1}|\mathcal{F}_{h,1}^{k+1}] = 0$ holds. Notice that $t(k,0,2) = t(k-1,H,2)$ and $\mathcal{F}_{0,2}^k = \mathcal{F}_{H,2}^{k-1}$ for $\forall k \ge 2$. Therefore, we can define the martingale sequence which is adapted to the filtration $\{\mathcal{F}_{h,m}^k\}_{(k,h,m)\in[K]\times[H]\times[2]}$,

$$S_{h,j}^{k+1} = \sum_{k'=1}^{k}\sum_{h'=0}^{H-1}\left(D_{h',1}^{k'} + D_{h',2}^{k'}\right) + \sum_{h'=0}^{h}\left(D_{h',1}^{k+1} + D_{h',2}^{k+1}\right) + \sum_{(k',h',j)\in[K]\times[H]\times[2]} D_{h',j}^{k'}$$

Then let

$$\sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\left(D_{h,1}^{k+1} + D_{h,2}^{k+1}\right) = S_{H,2}^{K-1}$$

Since $\gamma^h \widehat{Q}_h^{\widehat{\pi}^{k+1},k+1}, \gamma^{h+1}\widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1} \in [0, \hat{r}_{\max}/(1-\gamma))]$ and $\gamma^h Q_h^{\widehat{\pi}^{k+1},k+1}, \gamma^{h+1} V_{h+1}^{\widehat{\pi}^{k+1},k+1} \in [0, r_{\max}/(1-\gamma)]$. Then $|D_{h,1}^{k+1}|$, $|D_{h,s}^{k+1}| \le (r_{\max} \vee \hat{r}_{\max})/(1-\gamma)$ holds for $\forall (k,h) \in [K-1] \times [H]$. Then, by the Azuma-Hoeffding inequlaity, the following inequality holds.

$$\mathbb{P}\left(|S_{H,2}^{K-1}| \le s\right) \ge 2\exp\left(\frac{-s^2}{16\left(\frac{r_{\max}\vee\hat{r}_{\max}}{1-\gamma}\right)^2 \cdot (K-1)H}\right)$$

For any $p \in (0,1)$, if we set $s = 4(r_{\max} \vee \hat{r}_{\max})(1-\gamma)^{-1}\sqrt{(K-1)H\log(4/p)}$, then the inequality holds with probability at least $1 - p/2$.

Finally, for given $p \in (0,1)$, the term ③ can be bounded as follows,

$$\begin{aligned}③ &= \sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\left(D_{h,1}^{k+1} + D_{h,2}^{k+1}\right) - \sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\gamma^h \iota_h^{k+1}(s_h^{k+1}, a_h^{k+1}) \\ &\le \frac{4(r_{\max} \vee \hat{r}_{\max})}{1-\gamma}\sqrt{(K-1)H\log(4/p)} - \iota_H^{K-1} \qquad \text{(D.35)}\end{aligned}$$

where we have defined $\iota_H^{K-1} = \sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\gamma^h \iota_h^{k+1}(s_h^{k+1}, a_h^{k+1})$

4. Upper bound of dynamic regret.

4.1. Upper bound of dynamic regret - without entropy regularization.

Finally, for the entropy-regularized case, combining the equations (D.21), (D.25), (D.35) gives us the upper bound of dynamic regret of future policy $\{\widehat{\pi}\}$ that holds with probability $1 - p/2$.

$$\begin{aligned}
&\mathfrak{R}\left(\{\widehat{\pi}^{k+1}\}_{1:K-1}, K)\right) \\
&= ① + ② + ③ \\
&\leq (K-1) \cdot \frac{\gamma^H}{1-\gamma}(r_{\max} + \hat{r}_{\max}) + \frac{1}{1-\gamma}\bar{\iota}_{\infty}^{K} \\
&+ (K-1)\left(\frac{1}{(1-\gamma)^2\Delta_\pi} + \frac{\log|\mathcal{A}|}{\eta\Delta_\pi} + \frac{2\gamma^H \widehat{r}_{\max}}{1-\gamma}\right) \\
&+ \frac{4(r_{\max} \vee \hat{r}_{\max})}{1-\gamma}\sqrt{(K-1)H\log(4/p)} - \iota_H^{K-1}
\end{aligned}$$

Taking a upper bound of $r_{\max},\ \hat{r}_{\max}$ using $(r_{\max} \vee \hat{r}_{\max})$ yields the following final upper bound that holds with probability $1 - p/2$,

$$\begin{aligned}
&\mathfrak{R}\left(\{\widehat{\pi}^{k+1}\}_{1:K-1}, K)\right) \\
&\leq (K-1)\left(\frac{1}{(1-\gamma)^2\Delta_\pi} + \frac{\log|\mathcal{A}|}{\eta\Delta_\pi} + \frac{4\gamma^H(\widehat{r}_{\max} \vee r_{\max})}{1-\gamma}\right. \\
&\left. + \frac{4(r_{\max} \vee \hat{r}_{\max})}{1-\gamma}\sqrt{\frac{H\log(4/p)}{K-1}}\right) + \frac{1}{1-\gamma}\bar{\iota}_{\infty}^{K} - \iota_H^{K-1}
\end{aligned}$$

4.2. Upper bound of dynamic regret - with entropy regularization.

Finally, for the entropy-regularized case, combining the equations (D.21), (D.33), (D.35) gives us the upper bound of dynamic regret of future policy $\{\widehat{\pi}\}$ that holds with probability at least $1 - p/2$.

$$\begin{aligned}
&\mathfrak{R}\left(\{\widehat{\pi}^{k+1}\}_{1:K-1}, K)\right) \\
&= ① + ② + ③ \\
&\leq (K-1) \cdot \frac{\gamma^H}{1-\gamma}(r_{\max} + \hat{r}_{\max}) + \frac{1}{1-\gamma}\bar{\iota}_{\infty}^{K} \\
&+ (K-1)\left((\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi - 1}C_1 + C_2\right] + \frac{2\gamma^H \widehat{r}_{\max}}{1-\gamma} + \frac{2\tau\log|\mathcal{A}|}{1-\gamma}\right) \\
&+ \frac{4(r_{\max} \vee \hat{r}_{\max})}{1-\gamma}\sqrt{(K-1)H\log(4/p)} - \iota_H^{K-1}
\end{aligned}$$

Taking a upper bound of $r_{\max},\ \hat{r}_{\max}$ using $(r_{\max} \vee \hat{r}_{\max})$ yields the following final upper bound that holds with probability at least $1 - p/2$,

$$\begin{aligned}
&\mathfrak{R}\left(\{\widehat{\pi}^{k+1}\}_{1:K-1}, K)\right) \\
&\leq (K-1)\left((\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi - 1}C_1 + C_2\right] + \frac{4\gamma^H(\widehat{r}_{\max} \vee r_{\max})}{1-\gamma} + \frac{2\tau\log|\mathcal{A}|}{1-\gamma}\right. \\
&\left. + \frac{4(r_{\max} \vee \hat{r}_{\max})}{1-\gamma}\sqrt{\frac{H\log(4/p)}{K-1}}\right) + \frac{1}{1-\gamma}\bar{\iota}_{\infty}^{K} - \iota_H^{K-1}
\end{aligned}$$

Then, combining **4.1**, **4.2** provides the following expression,

$$\mathfrak{R}\left(\{\widehat{\pi}^{k+1}\}_{1:K-1}, K)\right) \leq \mathfrak{R}_{\mathcal{I}} + \mathfrak{R}_{✠}$$

where

$$\mathfrak{R}_{\mathcal{I}} = \frac{1}{1-\gamma}\bar{\iota}_{\infty}^{K} - \iota_{H}^{K} + C_p\sqrt{K-1}$$
$$\mathfrak{R}_{\texttt{Alg}} = C_{\texttt{Alg}}(\Delta_\pi)\cdot(K-1)$$
$$\mathfrak{R}_{\texttt{Alg}_\tau} = C_{\texttt{Alg}_\tau}(\Delta_\pi)\cdot(K-1)$$

and corresponding constants are

$$C_p = \frac{4(r_{\max}\vee\hat{r}_{\max})}{1-\gamma}\sqrt{H\log(4/p)},\quad C_{\texttt{Alg}}(\Delta_\pi) = \left(\frac{1}{(1-\gamma)^2} + \frac{\log|\mathcal{A}|}{\eta}\right)\cdot\frac{1}{\Delta_\pi} + \frac{4\gamma^H(\widehat{r}_{\max}\vee r_{\max})}{1-\gamma}$$
$$C_{\texttt{Alg}_\tau}(\Delta_\pi) = (\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi-1}C_1 + C_2\right] + \frac{4\gamma^H(\widehat{r}_{\max}\vee r_{\max})}{1-\gamma} + \frac{2\tau\log|\mathcal{A}|}{1-\gamma}$$

□

Lemma 1 (Conditions on Δ_π, H to guarantee the optimal threshold 2ϵ of ② without entropy regularization). *Let us estimate the conditions on $\tau, \eta, \Delta_\pi, H$ when the each terms of RHS of the equation* (D.32) *is bounded by ϵ as follows,*

$$②\text{'s } (k)^{th} \text{ term} = \underbrace{\frac{1}{(1-\gamma)^2\Delta_\pi} + \frac{\log|\mathcal{A}|}{\eta\Delta_\pi}}_{②\text{-}ⓐ\,\le\,\epsilon} + \underbrace{\frac{2\gamma^H\widehat{r}_{max}}{1-\gamma}}_{②\text{-}ⓑ\,\le\,\epsilon}$$

The term ② − ⓐ, ⓑ provides the following conditions,

$$②-ⓐ : \Delta_\pi \ge \left(\frac{1}{(1-\gamma)^2} + \frac{\log|\mathcal{A}|}{\eta}\right)\cdot\frac{1}{\epsilon}$$
$$②-ⓑ : H \ge \frac{\log(\frac{1-\gamma}{2\widehat{r}_{max}}\epsilon)}{\log(\gamma)}$$
$$: H \ge \frac{1}{1-\gamma}\log\left(\frac{2\widehat{r}_{max}}{(1-\gamma)\epsilon}\right)$$

Lemma 2 (Conditions on τ, Δ_π, H to guarantee the optimal threshold 4ϵ of ② with entropy regularization). *Let us estimate the conditions on $\tau, \eta, \Delta_\pi, H$ when the each terms of RHS of the equation* (D.32) *is bounded by ϵ as follows,*

$$②\text{'s } (k)^{th} \text{ term} = \underbrace{(\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi-1}C_1\right]}_{②\text{-}ⓐ\,\le\,\epsilon} + \underbrace{(\gamma+2)\,C_2}_{②\text{-}ⓑ\,\le\,\epsilon} + \underbrace{\frac{2\gamma^H\widehat{r}_{max}}{1-\gamma}}_{②\text{-}ⓒ\,\le\,\epsilon} + \underbrace{\frac{2\tau\log|\mathcal{A}|}{1-\gamma}}_{②\text{-}ⓓ\,\le\,\epsilon}$$

The term ② − ⓑ, ⓒ, ⓓ provides the following conditions as follows,

$$②-ⓑ : \delta \le \frac{\epsilon}{(\gamma+2)\cdot\frac{2}{1-\gamma}\cdot(1+\frac{\gamma}{\eta\tau})} \tag{D.36}$$
$$②-ⓒ : H \ge \frac{\log(\frac{1-\gamma}{2\widehat{r}_{max}}\epsilon)}{\log(\gamma)}$$
$$: H \ge \frac{1}{1-\gamma}\log\left(\frac{2\widehat{r}_{max}}{(1-\gamma)\epsilon}\right) \tag{D.37}$$
$$②-ⓓ : \tau \le \frac{1-\gamma}{2\log|\mathcal{A}|}\epsilon \tag{D.38}$$

and the term ② − ⓐ offers the lower bound of iteration Δ_π as follows.

$$\begin{aligned} ② - ⓐ : \Delta_\pi &\geq \frac{\log\left(\frac{\epsilon}{C_1(\gamma+2)}\right)}{\log(1-\eta\tau)} + 1 \\ : \Delta_\pi &\geq \frac{1}{\eta\tau}\log\left(\frac{C_1(\gamma+2)}{\epsilon}\right) + 1 \end{aligned} \tag{D.39}$$

The equation (D.37), (D.39) holds by applying first order Taylor series on $\log(\gamma),\ \log(1-\eta\tau)$ *since* $\gamma \in (0,1],\ \eta \in (0,(1-\gamma)/\tau]$. *The equation (D.36), (D.39) implies that if the learning rate* η *is fixed in the admissible range, then the iteration complexity scales inversely proportional with* τ, *and the approximation gap bound (*δ_{max}*) also scales proportional with* τ.

Now, the best convergence guaranteed is achieved when $\eta^* = (1-\gamma)/\tau$ *(The* η *that minimizes the equation (D.29)), the condition of hyperparameters* $\Delta_{\pi,\eta^*}, \delta_{\eta^*}$ *are given as follows,*

$$\begin{aligned} ② - ⓐ &: \Delta_{\pi,\eta^*} \geq \frac{1}{1-\gamma}\log\left(\frac{\|\widehat{Q}^{*,k+1}_\tau - \widehat{Q}^{\widehat{\pi}^{(0)}}_\tau\|_\infty(\gamma+2)}{\epsilon}\right) + 1 \\ ② - ⓑ &: \delta_{\eta^*} \leq \frac{\epsilon(1-\gamma)^2}{2(\gamma+2)} \end{aligned}$$

When $\eta^* = (1-\gamma)/\tau$, *the iteration complexity is now proportional to the effective horizon* $1/(1-\gamma)$ *modulo some log factor, and iteration complexity and* δ_{max} *is now independent of the choice of the regularization parameter* τ

Lemma 3 (**Sample complexity to guarantee the optimal threshold** 4ϵ **of ②**). *The authors of the paper [40] showed that for any policy* π, $\|\widehat{Q}^{\widehat{\pi}^{(\Delta_\pi)}}_\tau - \widehat{Q}^{\widehat{\pi}^{(\Delta_\pi)}}_\tau\|_\infty \leq \delta$ *holds with high probability with model-based policy evaluation, as long as the number of samples per state-action pairs exceeds the order of* $1/((1-\gamma)^3\delta^2)$ *up to some logarithmic factor.*

Then, set δ_{max} *as RHS of the equation (D.36), then if we have the number of samples per state-action pairs at least the order of*

$$\frac{1}{(1-\gamma)^3\delta^2_{max}}$$

up to some logarithmic factor, then the $\delta \leq \delta_{max}$ *holds with high probability, and we can guarantee the optimal threshold* 4ϵ *with high probability if all corresponding conditions, equation (D.37), (D.38), (D.39), holds.*

Proof of Theorem 2. **1. ProST-T** ι^K_H **:**

The *empirical* estimated model prediction error $\iota^{k+1}_h(s^{k+1}_h, a^{k+1}_h)$ is represented as follows (Definition (C.3)).

$$-\iota^{k+1}_h(s^{k+1}_h, a^{k+1}_h) = -R^{(k+1)}(s^{k+1}_h, a^{k+1}_h) - \gamma(P^{(k+1)}\widehat{V}^{\widehat{\pi}^{k+1},k+1}_{h+1})(s^{k+1}_h, a^{k+1}_h) + \widehat{Q}^{\widehat{\pi}^{k+1},k+1}_h(s^{k+1}_h, a^{k+1}_h) \tag{D.40}$$

$$\begin{aligned} &= -R^{(k+1)}(s^{k+1}_h, a^{k+1}_h) - \gamma(P^{(k+1)}\widehat{V}^{\widehat{\pi}^{k+1},k+1}_{h+1})(s^{k+1}_h, a^{k+1}_h) \\ &\quad + \widehat{R}^{(k+1)}(s^{k+1}_h, a^{k+1}_h) + \gamma(\widehat{P}^{(k+1)}\widehat{V}^{\widehat{\pi}^{k+1},k+1}_{h+1})(s^{k+1}_h, a^{k+1}_h) \qquad \text{(D.41)} \\ &= \left(\widehat{R}^{(k+1)} - R^{(k+1)}\right)(s^{k+1}_h, a^{k+1}_h) + \gamma\left(\left(\widehat{P}^{(k+1)} - P^{(k+1)}\right)\widehat{V}^{\widehat{\pi}^{k+1},k+1}_{h+1}\right)(s^{k+1}_h, a^{k+1}_h) \\ &\leq \bar{\Delta}^{Bonus,r}_{k,h} + \gamma\left\|\left(\widehat{P}^{(k+1)} - P^{(k+1)}\right)(\cdot \mid s^{k+1}_h, a^{k+1}_h)\right\|_1 \left\|\widehat{V}^{\widehat{\pi}^{k+1},k+1}_{h+1}(\cdot)\right\|_\infty \\ &\leq \bar{\Delta}^{Bonus,r}_{k,h} + \gamma\bar{\Delta}^{p}_{k,h}\frac{\gamma^{H-h}\hat{r}_{\max}}{1-\gamma} \qquad \text{(D.42)} \\ &\leq \bar{\Delta}^{r}_{k,h} + 2\Gamma^k_w(s^{k+1}_h, a^{k+1}_h) + \gamma\bar{\Delta}^{p}_{k,h}\frac{\gamma^{H-h}\hat{r}_{\max}}{1-\gamma} \qquad \text{(D.43)} \end{aligned}$$

The equation (D.41) holds due to the future bellman equation (C.8), the equation (D.42) holds since $\left\|\widehat{V}_{h+1}^{\widehat{\pi}^{k+1},k+1}(\cdot)\right\|_\infty \le \sum_{h'=h+1}^{H}\gamma^{h'-(h+1)}\hat{r}_{\max} \le \gamma^{H-h}\hat{r}_{\max}/(1-\gamma)$, and the equation (D.43) holds since $\Delta_{r,k}^{Bonus,1}(s,a) \le \left|\left(R^{(k+1)} - \widetilde{R}^{(k+1)}\right)(s,a)\right| + |2\Gamma_w^k(s,a)| = \Delta_{r,k}^1(s,a) + 2\Gamma_w^k(s,a)$ holds for (s,a). Then the summation of pseudo empirical model prediction error over all episodes and all steps is bounded as follows,

$$-\iota_H^K = \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} -\gamma^h \iota_h^{k+1}(s_h^{k+1},a_h^{k+1}) \le \underbrace{\bar{\Delta}_K^r}_{①} + \underbrace{\sum_{k=1}^{K-1}\sum_{h=0}^{H-1} 2\Gamma_w^k(s_h^{k+1},a_h^{k+1})}_{②} + \frac{\gamma\hat{r}_{\max}}{1-\gamma}\underbrace{\bar{\Delta}_K^p}_{③} \tag{D.44}$$

We use the Lemma 8 to bound the term ①, Lemma 9 and equation (D.13) to bound the term ②, and Lemma 11(or Lemma 10) to bound the term ③. Then, each terms can be finally bounded as

$$① \le wHB_r(\Delta_\pi) + \lambda r_{\max}\cdot(K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} \tag{D.45}$$

$$② \le 2\beta(K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} \tag{D.46}$$

$$③ \le \left(|\mathcal{S}|\sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda\right)(K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} + wHB_p(\Delta_\pi) \tag{D.47}$$

where equation (D.47) holds with probability $1-\delta$, $\delta \in (0,1)$.Now, combine (D.45), (D.46), (D.47) yields the following results.

$$\begin{aligned}
-\iota_H^K =& -\sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\iota_h^{k+1}(s_h^{k+1},a_h^{k+1}) \\
\le& \underbrace{\bar{\Delta}_K^r}_{①} + \underbrace{\sum_{k=1}^{K-1}\sum_{h=0}^{H-1} 2\Gamma_w^k(s_h^{k+1},a_h^{k+1})}_{②} + \frac{\gamma\hat{r}_{\max}}{1-\gamma}\underbrace{\bar{\Delta}_K^p}_{③} \\
\le& wHB_r(\Delta_\pi) + \lambda r_{\max}\cdot(K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} + 2\beta(K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} \\
&+ \frac{\gamma\hat{r}_{\max}}{1-\gamma}\left(\left(|\mathcal{S}|\sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda\right)(K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} + wHB_p(\Delta_\pi)\right) \\
\le& wH\left(B_r(\Delta_\pi) + \frac{\gamma\hat{r}_{\max}}{1-\gamma}B_p(\Delta_\pi)\right) \\
&+ (K-1)\sqrt{H}\left(\lambda r_{\max} + 2\beta + \frac{\gamma\hat{r}_{\max}}{1-\gamma}\left(|\mathcal{S}|\sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda\right)\right)\sqrt{\frac{1}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)}
\end{aligned} \tag{D.48}$$

2. ProST-T $\bar{\iota}_\infty^K$:

Recall that $\bar{\iota}_\infty^K = \sum_{k=1}^{K-1} \bar{\iota}_\infty^{k+1}$. For same δ that we used in the previous proof of [1.ProST-T ι_H^K] (see equation (D.48)), $\bar{\iota}_\infty^k$ can be bounded as follows with probability $1-\delta$.

$$
\begin{aligned}
\bar{\iota}_\infty^{k+1} &= R^{(k+1)} + \gamma P^{(k+1)} \widehat{V}_\infty^{*,k+1} - \widehat{Q}_\infty^{*,k+1} \\
&= R^{(k+1)} + \gamma P^{(k+1)} \widehat{V}_\infty^{*,k+1} - \left(\widehat{R}^{(k+1)} + \gamma \widehat{P}^{(k+1)} \widehat{V}_\infty^{*,k+1} \right) && \text{(D.49)} \\
&= R^{(k+1)} + \gamma P^{(k+1)} \widehat{V}_\infty^{*,k+1} - \left(\widetilde{R}^{(k+1)} + 2\Gamma_w^k(s,a) + \gamma \widehat{P}^{(k+1)} \widehat{V}_\infty^{*,k+1} \right) && \text{(D.50)} \\
&= R^{(k+1)} + \gamma P^{(k+1)} \widehat{V}_\infty^{*,k+1} - \left(\widetilde{R}^{(k+1)} + 2\beta(\Lambda_w^k(s,a))^{1/2} + \gamma \widehat{P}^{(k+1)} \widehat{V}_\infty^{*,k+1} \right) && \text{(D.51)} \\
&= \left(R^{(k+1)} - \widetilde{R}^{(k+1)} \right) - \beta(\Lambda_w^k(s,a))^{1/2} + \gamma \left(P^{(k+1)} - \widehat{P}^{(k+1)} \right) \widehat{V}_\infty^{*,k+1} - \beta(\Lambda_w^k(s,a))^{1/2} && \text{(D.52)} \\
&\le |R^{(k+1)} - \widetilde{R}^{(k+1)}| - \beta(\Lambda_w^k(s,a))^{1/2} + \gamma \|P^{(k+1)} - \widehat{P}^{(k+1)}\|_1 \|\widehat{V}_\infty^{*,k+1}\|_\infty - \beta(\Lambda_w^k(s,a))^{1/2} \\
&\le \left(B_{r,w}^k(\Delta_\pi) + \lambda \Lambda_w^k(s,a) r_{\max} \right) - \beta(\Lambda_w^k(s,a))^{1/2} && \text{(D.53)} \\
&+ \gamma \cdot \left(B_{p,w}^k(\Delta_\pi) + (\Lambda_w^k(s,a))^{1/2} \cdot |\mathcal{S}| \cdot \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta\lambda}\right)} + \lambda \Lambda_w^k(s,a) \right) \cdot \frac{\hat{r}_{\max}}{1-\gamma} - \beta(\Lambda_w^k(s,a))^{1/2} && \text{(D.54)} \\
&\le \left(B_{r,w}^k(\Delta_\pi) + \lambda (\Lambda_w^k(s,a))^{1/2} r_{\max} \right) - \beta(\Lambda_w^k(s,a))^{1/2} && \text{(D.55)} \\
&+ \gamma \cdot \left(B_{p,w}^k(\Delta_\pi) + (\Lambda_w^k(s,a))^{1/2} \cdot |\mathcal{S}| \cdot \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta\lambda}\right)} + \lambda (\Lambda_w^k(s,a))^{1/2} \right) \cdot \frac{\hat{r}_{\max}}{1-\gamma} - \beta(\Lambda_w^k(s,a))^{1/2} && \text{(D.56)} \\
&\le B_{r,w}^k(\Delta_\pi) + \gamma B_{p,w}^k(\Delta_\pi) + \underbrace{\left(\lambda r_{\max} - \beta + \gamma |\mathcal{S}| \cdot \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta\lambda}\right)} + \frac{\lambda \hat{r}_{\max}}{1-\gamma} - \beta \right)}_{\le 0} (\Lambda_w^k(s,a))^{1/2} && \text{(D.57)} \\
&\le B_{r,w}^k(\Delta_\pi) + \gamma B_{p,w}^k(\Delta_\pi) && \text{(D.58)}
\end{aligned}
$$

The equation (D.49) holds by the future bellman equation (C.9) when $h = \infty$, the equations (D.50) and (D.51) hold by the definition of $\widehat{R}^{(k+1)}$, equation (D.13). The equations (D.53), (D.54) hold by lemma 7, lemma 10, equation (D.8), equation (D.9). The equations (D.55), (D.56) hold since $0 \le \Lambda_w^k(s,a) < 1$. Now, the equation (D.58) holds *if* the under-brace term of equation (D.57) is equal or smaller than zero. That gives us the additional following condition of β to obtain the final inequality (D.58). Recall that $\hat{r}_{\max}$ is defined as $\tilde{r}_{\max} + \frac{2\beta}{\sqrt{\lambda}}$ where $\widetilde{r}_{\max}$ is a constant, but $\hat{r}_{\max}$ is still function of β, λ (equation (D.14)).

$$
\lambda r_{\max} - \beta + \gamma |\mathcal{S}| \cdot \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta\lambda}\right)} + \frac{\lambda}{1-\gamma} \cdot \left(\widetilde{r}_{\max} + \frac{2\beta}{\sqrt{\lambda}} \right) - \beta \le 0
$$

solving above condition yield the condition of β .

$$
\beta \ge \left(2 + \frac{2\sqrt{\lambda}}{1-\gamma} \right)^{-1} \left(\lambda r_{\max} + \gamma |\mathcal{S}| \cdot \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta\lambda}\right)} \right) \tag{D.59}
$$

Since equation (D.58) holds for all (s,a), if β satisfies the equation (D.59), then $\sum_{k=1}^{K-1} \bar{\iota}_\infty^K = \|\bar{\iota}_\infty^k\|_\infty$ is bounded as

$$
\bar{\iota}_\infty^K \le \sum_{k=1}^{K-1} \left(B_{r,w}^k(\Delta_\pi) + \gamma B_{p,w}^k(\Delta_\pi) \right) \le w(B_r(\Delta_\pi) + \gamma B_p(\Delta_\pi))
$$

because $\sum_{k=1}^{K-1} B_{p,w}^k(\Delta_\pi) = \sum_{\mathcal{E}=1}^{\lfloor \frac{K-1}{w} \rfloor} \sum_{k=(\mathcal{E}-1)w}^{\mathcal{E}w} B_{p,w}^k(\Delta_\pi) = w B_p(\Delta_\pi)$ and same as B_r

Then, the model prediction errors $-\iota_H^K, \bar{\iota}_\infty^K$ when utilizing the forecaster f as `SW-LSE` are

$$
\begin{aligned}
-\iota_H^K \leq & wH\left(B_r(\Delta_\pi) + \frac{\gamma\hat{r}_{\max}}{1-\gamma}B_p(\Delta_\pi)\right) \\
& + (K-1)\sqrt{H}\left(\lambda r_{\max} + 2\beta + \frac{\gamma\hat{r}_{\max}}{1-\gamma}\left(|\mathcal{S}|\sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda\right)\right)\sqrt{\frac{1}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} \\
\bar{\iota}_\infty^K \leq & w(B_r(\Delta_\pi) + \gamma B_p(\Delta_\pi))
\end{aligned}
$$

Finally, the upper bound of $\mathfrak{R}_\mathcal{I}$ can be bounded as follows,

$$
\begin{aligned}
\mathfrak{R}_\mathcal{I} &= \frac{1}{1-\gamma}\bar{\iota}_\infty^K - \iota_H^K + C_p\sqrt{K-1} \\
&\leq \frac{1}{1-\gamma}\left(w(B_r(\Delta_\pi) + \gamma B_p(\Delta_\pi))\right) + wH\left(B_r(\Delta_\pi) + \frac{\gamma\hat{r}_{\max}}{1-\gamma}B_p(\Delta_\pi)\right) \\
&\quad + (K-1)\sqrt{H}\left(\lambda r_{\max} + 2\beta + \frac{\gamma\hat{r}_{\max}}{1-\gamma}\left(|\mathcal{S}|\sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda\right)\right)\sqrt{\frac{1}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} \\
&\quad + C_p\sqrt{K-1} \\
&\leq \left(\left(\frac{1}{1-\gamma} + H\right)B_r(\Delta_\pi) + \frac{(1+H\hat{r}_{\max})\gamma}{1-\gamma}B_p(\Delta_\pi)\right)w \\
&\quad + (K-1)\sqrt{H}\left(\lambda r_{\max} + 2\beta + \frac{\gamma\hat{r}_{\max}}{1-\gamma}\left(|\mathcal{S}|\sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda\right)\right)\sqrt{\frac{1}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} \\
&\quad + C_p\sqrt{K-1}
\end{aligned}
$$

Now, we let the $B(\Delta_\pi)$ as the conic combination of $B_r(\Delta_\pi), B_p(\Delta_\pi)$ as follows.

$$
\begin{aligned}
B(\Delta_\pi) &= \left(\frac{1}{1-\gamma} + H\right)B_r(\Delta_\pi) + \frac{(1+H\hat{r}_{\max})\gamma}{1-\gamma}B_p(\Delta_\pi) \\
&= \left(\frac{1}{1-\gamma} + H\right)\Delta_\pi^{\alpha_r}B_r(1) + \frac{(1+H\hat{r}_{\max})\gamma}{1-\gamma}\Delta_\pi^{\alpha_p}B_p(1) \\
&= C_{B_r}\Delta_\pi^{\alpha_r} + C_{B_p}\Delta_\pi^{\alpha_p}
\end{aligned}
$$

where we denote $C_{B_r} = \left(\frac{1}{1-\gamma} + H\right)B_r(1)$, $C_{B_p} = \frac{(1+H\hat{r}_{\max})\gamma}{1-\gamma}B_p(1)$ where those are constants related with total variation budget with reward and transition probability, respectively.

Recall the definition of $B_r(\Delta_\pi), B_p(\Delta_\pi)$ and $B_r(\Delta_\pi) = \Delta_\pi^{\alpha_r}B_r(1), B_p(\Delta_\pi) = \Delta_\pi^{\alpha_p}B_p(1)$ holds. We denote $B_p(1), B_r(1)$ as time-elapsing variation budget for one policy iteration. We also let the constant C_k to be defined as follows.

$$
C_k = (K-1)\sqrt{H}\left(\lambda r_{\max} + 2\beta + \frac{\gamma\hat{r}_{\max}}{1-\gamma}\left(|\mathcal{S}|\sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda\right)\right)
$$

Then, the upper bound of $\mathfrak{R}_\mathcal{I}$ is represented as follows.

$$
\mathfrak{R}_\mathcal{I} \leq B(\Delta_\pi)w + C_k\sqrt{\frac{1}{w}\log\left(\frac{\lambda+wH}{\lambda}\right)} + C_p\sqrt{K-1}
$$

□

Proof of Proposition 2. Now, let's set the sliding window length w which is adaptive to Δ_π as follows.

$$
\widetilde{w}(\Delta_\pi) = \left(\frac{C_k}{B(\Delta_\pi)}\right)^{2/3}
$$

Then, we have

$$B(\Delta_\pi)\widetilde{w}(\Delta_\pi) + C_k\sqrt{\frac{1}{\widetilde{w}(\Delta_\pi)}}\sqrt{\log\left(\frac{\lambda + \widetilde{w}(\Delta_\pi)H}{\lambda}\right)}$$
$$= C_k^{2/3}B(\Delta_\pi)^{1/3} + C_k^{2/3}B(\Delta_\pi)^{1/3}\sqrt{\log\left(1 + \frac{H}{\lambda}\left(\frac{C_k}{B(\Delta_\pi)}\right)^{2/3}\right)}$$

Since C_k is linear to $K-1$, the function $\mathfrak{R}_\mathcal{I}$ satisfies the following

$$\mathfrak{R}_\mathcal{I} = \mathcal{O}\left(B(\Delta_\pi)^{1/3}(K-1)^{2/3}\cdot\sqrt{\log\left(\frac{K-1}{B(\Delta_\pi)}\right)}\right) \tag{D.60}$$

Now, if $B(\Delta_\pi) = o(K)$, then $\mathfrak{R}_\mathcal{I}$ is sublinear to K. The corresponding condition is $B_r(1) + \frac{\hat{r}_{\max}}{1-\gamma}B_p(1) = o(K)$ and $G < K$ since

$$\begin{aligned}
&C_{B_r}\Delta_\pi^{\alpha_r} + C_{B_p}\Delta_\pi^{\alpha_p} = o(K)\\
&\left(C_{B_r} + C_{B_p}\right)\cdot\Delta_\pi^{\max(\alpha_r,\alpha_p)} = o(K)\\
&\left(\left(\frac{1}{1-\gamma} + H\right)B_r(1) + \left(\frac{1 + H\hat{r}_{\max}}{1-\gamma} +\right)B_p(1)\right)\cdot\Delta_\pi^{\max(\alpha_r,\alpha_p)} = o(K)\\
&\left(\frac{1}{1-\gamma}\left(B_r(1) + B_p(1)\right) + H\left(B_r(1) + \frac{\hat{r}_{\max}}{1-\gamma}B_p(1)\right)\right)\cdot\Delta_\pi^{\max(\alpha_r,\alpha_p)} = o(K)
\end{aligned}$$

holds. □

Proof of Theorem 3. We prove the followings,

1. $\mathfrak{R}_{\mathtt{Alg}_\tau}(\Delta_\pi)$ is a non-increasing function, $\mathfrak{R}_\mathcal{I}(\Delta_\pi)$ is an non-decreasing function , and both are convex function in a region $\mathcal{G}_{\mathtt{Alg}_\tau}\cap\mathcal{G}_\mathcal{I}$

$$\begin{aligned}
\frac{\partial\mathfrak{R}_{A_\tau}(\Delta_\pi)}{\partial\Delta_\pi} &= \frac{\partial}{\partial\Delta_\pi}\left(C_1(K-1)(\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi-1}\right]\right)\\
&= \log(1-\eta\tau)C_1(K-1)(\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi-1}\right] \le 0\\
\frac{\partial^2\mathfrak{R}_{A_\tau}(\Delta_\pi)}{\partial^2\Delta_\pi} &= \frac{\partial^2}{\partial^2\Delta_\pi}\left(C_1(K-1)(\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi-1}\right]\right)\\
&= \left(\log(1-\eta\tau)\right)^2 C_1(K-1)(\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi-1}\right] \ge 0
\end{aligned}$$

since $G \in \mathcal{G}_{\mathtt{Alg}_\tau}\cap\mathcal{G}_\mathcal{I}$ satisfies $G > 1$ and $\log(1-\eta\tau) \le 0$ holds from the hyperparameter assumption $0 \le \eta \le (1-\gamma)/\tau$ from the Proposition 1.

$$\begin{aligned}
\frac{\partial\mathfrak{R}_\mathcal{I}(\Delta_\pi)}{\partial\Delta_\pi} &= \frac{\partial}{\partial\Delta_\pi}\left(C_{B_r}\Delta_\pi^{\alpha_r} + C_{B_p}\Delta_\pi^{\alpha_p}\right)\\
&= \alpha_r C_{B_r}\Delta_\pi^{\alpha_r-1} + \alpha_p C_{B_p}\Delta_\pi^{\alpha_p-1} \ge 0\\
\frac{\partial^2\mathfrak{R}_\mathcal{I}(\Delta_\pi)}{\partial^2\Delta_\pi} &= \frac{\partial^2}{\partial^2\Delta_\pi}\left(C_{B_r}\Delta_\pi^{\alpha_r} + C_{B_p}\Delta_\pi^{\alpha_p}\right)\\
&= \alpha_r(\alpha_r-1)C_{B_r}\Delta_\pi^{\alpha_r-2} + \alpha_p(\alpha_p-1)C_{B_p}\Delta_\pi^{\alpha_p-2} \ge 0
\end{aligned}$$

holds only $\alpha_r, \alpha_p \ge 1$.

2. Optimal $G^*_{\mathtt{Alg}}, G^*_{\mathtt{Alg}_\tau}$ for $\mathtt{Alg}, \mathtt{Alg}_\tau$

For the optimal G, we slightly relax $\mathfrak{R}_\mathcal{I}(\Delta_\pi) = C_{B_r}\Delta_\pi^{\alpha_r} + C_{B_p}\Delta_\pi^{\alpha_p}$ to $\mathfrak{R}'_\mathcal{I}(\Delta_\pi) = \left(C_{B_r} + C_{B_p}\right)\Delta_\pi^{\max(\alpha_r,\alpha_p)}$ and get the optimal Δ^*_π of the dynamic regret $\mathfrak{R}_{\text{✠}}(\Delta_\pi) + \mathfrak{R}'_\mathcal{I}(\Delta_\pi)$. We omit the proof of that $\mathfrak{R}'_\mathcal{I}(\Delta_\pi)$ is also non-decreasing function (simliar process to 1.).

1. $\max(\boldsymbol{\alpha_r}, \boldsymbol{\alpha_p}) = 0$: this means $\mathfrak{R}_{\mathcal{I}}(\Delta_\pi) = C_{B_r} + C_{B_p}$ that $\mathfrak{R}_{\mathcal{I}}$ is now independent to G. Then, for both baseline algorithm $\texttt{Alg}, \texttt{Alg}_\tau$, infinite number of G guarantees small dynamic regret $\mathfrak{R}_{\mathcal{I}}$, that leads to small $\mathfrak{R}$. We can check $\mathfrak{R}_{\texttt{Alg}}$ decreases in a $1/G$ scale, and $\mathfrak{R}_{\texttt{Alg}_\tau}$ decreases in a $\exp(\Delta_\pi)$ scale. So using entropy-regularization guarantees faster convergence.

For the remaining case, let's first compute the gradient of the term $\mathfrak{R}_{\texttt{Alg}_\tau}(\Delta_\pi) + \mathfrak{R}'_{\mathcal{I}}(\Delta_\pi)$,

$$
\begin{aligned}
&\frac{\partial \mathfrak{R}_{A_\tau}(\Delta_\pi) + \mathfrak{R}'_{\mathcal{I}}(\Delta_\pi)}{\partial \Delta_\pi} \\
&= \max(\alpha_r, \alpha_p)\left(\alpha_r C_{B_r} + \alpha_p C_{B_p}\right)\Delta_\pi^{\max(\alpha_r,\alpha_p)-1} - \log\left(\frac{1}{1-\eta\tau}\right) C_1 (K-1)(\gamma+2)\left[(1-\eta\tau)^{\Delta_\pi - 1}\right] \\
&= k_B \Delta_\pi^{\max(\alpha_r,\alpha_p)-1} - k_{\texttt{Alg}_\tau}\left[(1-\eta\tau)^{\Delta_\pi - 1}\right]
\end{aligned}
$$

and the term $\mathfrak{R}_{\texttt{Alg}}(\Delta_\pi) + \mathfrak{R}'_{\mathcal{I}}(\Delta_\pi)$,

$$
\begin{aligned}
&\frac{\partial \mathfrak{R}_{A}(\Delta_\pi) + \mathfrak{R}'_{\mathcal{I}}(\Delta_\pi)}{\partial \Delta_\pi} \\
&= \max(\alpha_r, \alpha_p)\left(\alpha_r C_{B_r} + \alpha_p C_{B_p}\right)\Delta_\pi^{\max(\alpha_r,\alpha_p)-1} - \left(\frac{1}{(1-\gamma)^2} + \frac{\log|\mathcal{A}|}{\eta}\right)\cdot\frac{1}{\Delta_\pi^2} \\
&= k_B \Delta_\pi^{\max(\alpha_r,\alpha_p)-1} - k_{\texttt{Alg}}\frac{1}{\Delta_\pi^2}
\end{aligned}
$$

2. $\max(\boldsymbol{\alpha_r}, \boldsymbol{\alpha_p}) = \mathbf{1}$: Then $(1-\eta\tau)^{\Delta_\pi - 1} = k_B / k_{\texttt{Alg}_\tau}$ and $\Delta_\pi^{-2} = k_B / k_{\texttt{Alg}}$ should be satisfied, respectively. Then, the optimal $G^*_{\texttt{Alg}_\tau} = \log_{1-\eta\tau}(k_B/k_{\texttt{Alg}_\tau}) + 1$ and the optimal $G^*_{\texttt{Alg}} = \sqrt{k_{\texttt{Alg}}/k_B}$

Now, for the case of A_τ, if $k_{\texttt{Alg}_\tau} = (1-\eta\tau)k_B$ is satisfied, $\frac{\partial \mathfrak{R}_{\texttt{Alg}_\tau}(\Delta_\pi) + \mathfrak{R}'_{\mathcal{I}}(\Delta_\pi)}{\partial \Delta_\pi} = 0$ is equals to solving $\Delta_\pi^{\max(\alpha_r,\alpha_p)-1} = (1-\eta\tau)^G$. Now, using the lambert W function to get G as follows.

$$
\begin{aligned}
&\Delta_\pi^{\max(\alpha_r,\alpha_p)-1} = (1-\eta\tau)^G \\
&(\max(\alpha_r,\alpha_p) - 1)\log G = G\log(1-\eta\tau) \\
&\Delta_\pi^{-1}\cdot\log G = \frac{\log(1-\eta\tau)}{\max(\alpha_r,\alpha_p)-1} \\
&-\log G\cdot e^{-\log G} = -\frac{\log(1-\eta\tau)}{\max(\alpha_r,\alpha_p)-1} \\
&W\left[-\log G\cdot e^{-\log G}\right] = W\left[-\frac{\log(1-\eta\tau)}{\max(\alpha_r,\alpha_p)-1}\right] \\
&W\left[-\log G\cdot e^{-\log G}\right] = W\left[-\frac{\log(1-\eta\tau)}{\max(\alpha_r,\alpha_p)-1}\right] \\
&-\log G = W\left[-\frac{\log(1-\eta\tau)}{\max(\alpha_r,\alpha_p)-1}\right] \\
&\Delta^*_\pi = \exp\left(-W\left[-\frac{\log(1-\eta\tau)}{\max(\alpha_r,\alpha_p)-1}\right]\right) = \exp(-W[x])
\end{aligned}
$$

3. $\mathbf{0 < \max(\boldsymbol{\alpha_r}, \boldsymbol{\alpha_p}) < 1}$:

   - $\texttt{Alg}$: $\Delta^*_{\pi\,\texttt{Alg}} = (k_B/k_{\texttt{Alg}})^{1/(\max(\alpha_r,\alpha_p)+1)}$
   - $\texttt{Alg}_\tau$: Then $x = -\frac{\log(1-\eta\tau)}{\max(\alpha_r,\alpha_p)-1} < 0$, then small $|x|$ have large $-W(x) > 0$ value, which leads to large $G^*_{\texttt{Alg}_\tau}$.

4. $\max(\alpha_r, \alpha_p) > 1$:

   - $\texttt{Alg}$: $\Delta^*_{\pi_{\texttt{Alg}}} = (k_B/k_{\texttt{Alg}})^{1/(\max(\alpha_r,\alpha_p)+1)}$
   - $\texttt{Alg}_\tau$: Then $x > 0$ satisfies and $-W(x) < 0$ satisfied. Then $\Delta^*_\pi < 1$ which means 1 iteration is enough.

□

From the proof of theorem 2, we can come up with Lemma 4, Lemma 5, Lemma 6 that upperbound two model prediction errors $-\iota^k_h$, $\bar{\iota}^k_\infty$

Lemma 4 (Upper bound of $-\iota^{k+1}_h(s^{k+1}_h, a^{k+1}_h)$ by $\bar{\Delta}^1_{r,k,h},\ \bar{\Delta}^1_{p,k,h}$)**.**

$$-\iota^{k+1}_h(s^{k+1}_h, a^{k+1}_h) \le \bar{\Delta}^r_{k,h} + 2\Gamma^k_w(s,a) + \gamma\bar{\Delta}^p_{k,h}\frac{\gamma^{H-h}\hat{r}_{max}}{1-\gamma}$$

Proof of Lemma 4. The equations (D.40)~(D.43) in the theorem 2 replace the proof. □

Lemma 5 (Upper bound of $-\iota^{k+1}_h(s,a)$ by $\Delta^1_{r,k},\ \Delta^1_{p,k}$)**.** *For any $(s,a) \in \mathcal{S}\times\mathcal{A}$, the following holds.*

$$-\iota^{k+1}_h(s,a) \le \Delta^1_{r,k}(s,a) + \gamma\Delta^p_k(s,a)\frac{\gamma^{H-h}\hat{r}_{max}}{1-\gamma} + 2\Gamma^k_w(s,a)$$

Proof of Lemma 5.

$$\begin{aligned}
-\iota^{k+1}_h(s,a) =& -R^{(k+1)}(s,a) - \gamma(P^{(k+1)}\widehat{V}^{\widehat{\pi}^{k+1},k+1}_{h+1})(s,a) + \widehat{Q}^{\widehat{\pi}^{k+1},k+1}_h(s,a)\\
=& -R^{(k+1)}(s,a) - \gamma(P^{(k+1)}\widehat{V}^{\widehat{\pi}^{k+1},k+1}_{h+1})(s,a)\\
&+ \widehat{R}^{(k+1)}(s,a) + \gamma(\widehat{P}^{(k+1)}\widehat{V}^{\widehat{\pi}^{k+1},k+1}_{h+1})(s,a)\\
=& \left(\widehat{R}^{(k+1)} - R^{(k+1)}\right)(s,a) + \gamma\left(\left(\widehat{P}^{(k+1)} - P^{(k+1)}\right)\widehat{V}^{\widehat{\pi}^{k+1},k+1}_{h+1}\right)(s,a)\\
\le& \Delta^1_{r,k}(s,a) + 2\Gamma^k_w(s,a) + \gamma\left\|\left(\widehat{P}^{(k+1)} - P^{(k+1)}\right)(\cdot\mid s,a)\right\|_1\left\|\widehat{V}^{\widehat{\pi}^{k+1},k+1}_{h+1}(\cdot)\right\|_\infty\\
\le& \Delta^1_{r,k}(s,a) + 2\Gamma^k_w(s,a) + \gamma\Delta^p_k(s,a)\frac{\gamma^{H-h}\hat{r}_{\max}}{1-\gamma}
\end{aligned}$$

□

Lemma 6 (Upper bound of $\bar{\iota}^k_\infty$ by $\Delta^1_{r,k},\ \Delta^1_{p,k}$)**.** *For any $(s,a) \in \mathcal{S}\times\mathcal{A}$, the following holds.*

$$\bar{\iota}^{k+1}_\infty(s,a) \le \Delta^1_{r,k}(s,a) + \Delta^1_{p,k}(s,a)\frac{\gamma\hat{r}_{max}}{1-\gamma} - 2\Gamma^k_w(s,a)$$

Proof of Lemma 6. Starting from the equation (D.52),

$$\begin{aligned}
\bar{\iota}^{k+1}_\infty &= \left(R^{(k+1)} - \widetilde{R}^{(k+1)}\right) - \beta(\Lambda^k_w(s,a))^{1/2} + \gamma\left(P^{(k+1)} - \widehat{P}^{(k+1)}\right)\widehat{V}^{*,k+1}_\infty - \beta(\Lambda^k_w(s,a))^{1/2}\\
&\le \left|R^{(k+1)} - \widetilde{R}^{(k+1)}\right| - \beta(\Lambda^k_w(s,a))^{1/2} + \gamma\left\|P^{(k+1)} - \widehat{P}^{(k+1)}\right\|_1\left\|\widehat{V}^{*,k+1}_\infty\right\|_\infty - \beta(\Lambda^k_w(s,a))^{1/2}\\
&\le \Delta^1_{r,k}(s,a) - \beta(\Lambda^k_w(s,a))^{1/2} + \gamma\Delta^1_{p,k}(s,a)\frac{\hat{r}_{max}}{1-\gamma} - \beta(\Lambda^k_w(s,a))^{1/2}\\
&= \Delta^1_{r,k}(s,a) + \Delta^1_{p,k}(s,a)\frac{\gamma\hat{r}_{max}}{1-\gamma} - 2\Gamma^k_w(s,a)
\end{aligned}$$

□

Lemma 7 (Upper bound of $\Delta^1_{r,k}(s,a)$)**.** *For any $(s,a) \in \mathcal{S}\times\mathcal{A}$, $\Delta^1_{r,k}(s,a)$ is bounded as follows.*

$$\Delta^1_{r,k}(s,a) \le B^k_{r,w}(\Delta_\pi) + \lambda\Lambda^k_w(s,a)r_{max}$$

Proof of Lemma 7. We directly utilize the proof of the lemma (35) of the paper [32]. Then, for any $(s,a) \in \mathcal{S} \times \mathcal{A}$, $\Delta^1_{r,k}(s,a)$ can be represented as follows,

$$\Delta^1_{r,k}(s,a) \tag{D.61}$$
$$= |R^{(k+1)}(s,a) - \widetilde{R}^{(k+1)}(s,a)| \tag{D.62}$$
$$= |o^r_{(k+1)}(s,a) - \widetilde{o}^r_{(k+1)}(s,a)| \tag{D.63}$$
$$= \left| \frac{\sum_{t=(1\wedge k-w+1)}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot r_h^t}{\lambda + \sum_{t=(1\wedge k-w+1)}^{k} n_t(s,a)} - o^r_{(k+1)}(s,a) \right| \tag{D.64}$$
$$= \Lambda^k_w(s,a) \left| \sum_{t=(1\wedge k-w+1)}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot r_h^t - \left(\lambda + \sum_{t=(1\wedge k-w+1)}^{k} n_t(s,a)\right) o^r_{(k+1)}(s,a) \right| \tag{D.65}$$
$$= \Lambda^k_w(s,a) \left| \sum_{t=(1\wedge k-w+1)}^{k} \sum_{h=0}^{H-1} \left(\mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right]\left(r_h^t - o^r_{(k+1)}(s,a)\right)\right) - \lambda \cdot o^r_{(k+1)}(s,a) \right| \tag{D.66}$$
$$\leq \Lambda^k_w(s,a) \left(\sum_{t=(1\wedge k-w+1)}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot \left| r_h^t - o^r_{(k+1)}(s,a) \right| \right) + \lambda \Lambda^k_w(s,a) \left| o^r_{(k+1)}(s,a) \right| \tag{D.67}$$
$$\leq \Lambda^k_w(s,a) \left(\sum_{t=(1\wedge k-w+1)}^{k} n_t(s,a) \left(\left| r^t(s,a) - o^r_{(k+1)}(s,a) \right| \right) \right) + \lambda \Lambda^k_w(s,a) r_{\max} \tag{D.68}$$
$$\leq \max_{(1\wedge k-w+1) \leq t \leq k} \left(\left| r^t(s,a) - o^r_{(k+1)}(s,a) \right| \right) \Lambda^k_w(s,a) \left(\sum_{t=(1\wedge k-w+1)}^{k} n_t(s,a) \right) + \lambda \Lambda^k_w(s,a) r_{\max}$$
$$\leq \max_{(1\wedge k-w+1) \leq t \leq k} \left(\left| r^t(s,a) - o^r_{(k+1)}(s,a) \right| \right) + \lambda \Lambda^k_w(s,a) r_{\max}$$
$$\leq B^k_{r,w}(\Delta_\pi) + \lambda \Lambda^k_w(s,a) r_{\max} \tag{D.69}$$

Equations (D.63) and (D.64) hold by the definition of $o^r_{k+1}, \widetilde{o}^r_{k+1}$ (definition (D.7)), equation (D.65) holds by the definition (D.12), equation (D.66) holds since $n_t(s,a) \coloneqq \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right]$, equation (D.69) holds since $\max_{(1\wedge k-w+1) \leq t \leq k} \left(\left| r^t(s,a) - o^r_{(k+1)}(s,a) \right| \right) \leq |r^{(1\wedge k-w+1)}(s,a) - r^{(1\wedge k-w+1)+1}(s,a)| + \ldots + |R^{(k)}(s,a) - R^{(k+1)}(s,a)| = B^k_{r,w}(\Delta_\pi)$. □

Lemma 8 (Upper bound of $\bar{\Delta}^1_{r,K}$)**.** *For any $(s,a) \in \mathcal{S} \times \mathcal{A}$, the following holds,*

$$\bar{\Delta}^r_K \leq wHB_r(\Delta_\pi) + \lambda r_{max} \cdot (K-1) \sqrt{\frac{H}{w}} \sqrt{\log\left(\frac{\lambda + wH}{\lambda}\right)}$$

Proof of Lemma 8. The total empirical forecasting model error up to $K-1$ is given as

$$\begin{aligned}
\bar{\Delta}_K^r &= \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \bar{\Delta}_{k,h}^r \\
&= \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \bar{\Delta}_K^r(s_h^{k+1}, a_h^{k+1}) \\
&\le \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \left(B_{r,w}^k(\Delta_\pi) + \lambda\Lambda_w^k(s_h^{k+1}, a_h^{k+1}) r_{\max}\right) \qquad \text{(D.70)} \\
&= wHB_r(\Delta_\pi) + \lambda r_{\max} \cdot \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \left(\Lambda_w^k(s_h^{k+1}, a_h^{k+1})\right) \qquad \text{(D.71)} \\
&\le wHB_r(\Delta_\pi) + \lambda r_{\max} \cdot \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \left(\sqrt{\Lambda_w^k(s_h^{k+1}, a_h^{k+1})}\right) \\
&\le wHB_r(\Delta_\pi) + \lambda r_{\max} \cdot (K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda + wH}{\lambda}\right)} \qquad \text{(D.72)}
\end{aligned}$$

The equation (D.70) holds by lemma 7, the equation (D.71) holds since $\sum_{k=1}^{K-1} B_{r,w}^k(\Delta_\pi) = \sum_{\mathcal{E}=1}^{\lfloor \frac{K-1}{w} \rfloor} \sum_{k=(\mathcal{E}-1)w}^{\mathcal{E}w} B_{r,w}^k(\Delta_\pi) = wB_r(\Delta_\pi)$, the equation (D.72) holds by lemma 9. □

Lemma 9 (Upper bound of the term $\sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\sqrt{\Lambda_w^k(s_h^{k+1}, a_h^{k+1})}$)**.**

$$\sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \left(\sqrt{\Lambda_w^k(s_h^{k+1}, a_h^{k+1})}\right) \le (K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda + wH}{\lambda}\right)}$$

Proof of lemma 9. We denote $\bar{\Lambda}_w^k = \lambda\mathbb{I} + \sum_{t=(1\wedge k-w+1)}^{k}\sum_{h=0}^{H-1} \varphi(s_h^t, a_h^t)\varphi(s_h^t, a_h^t)^\top$. Also, we denote $(\bar{\Lambda}_w^k)^{(1)} = \lambda\mathbb{I} + \varphi(s_h^{(1\wedge k-w+1)}, a_h^{(1\wedge k-w+1)})\varphi(s_h^{(1\wedge k-w+1)}, a_h^{(1\wedge k-w+1)})^\top$ Then, for any $(s,a) \in \mathcal{S}\times\mathcal{A}$, $\Lambda_w^k(s,a) = \varphi(s,a)(\bar{\Lambda}_w^k)^{-1}\varphi(s,a)^\top$ holds.

Now, the following term can be bounded as follows.

$$\begin{aligned}
\sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\sqrt{\Lambda_w^k(s_h^{k+1}, a_h^{k+1})} &= \sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\sqrt{\varphi(s_h^{k+1}, a_h^{k+1})(\bar{\Lambda}_w^k)^{-1}\varphi(s_h^{k+1}, a_h^{k+1})^\top} \\
&= \sum_{\mathcal{E}=1}^{\lfloor \frac{K-1}{w} \rfloor} \sum_{k=(\mathcal{E}-1)w+1}^{\mathcal{E}w}\sum_{h=0}^{H-1}\sqrt{\varphi(s_h^{k+1}, a_h^{k+1})(\bar{\Lambda}_w^k)^{-1}\varphi(s_h^{k+1}, a_h^{k+1})^\top} \\
&\le \sum_{\mathcal{E}=1}^{\lfloor \frac{K-1}{w} \rfloor} \sqrt{Hw}\sqrt{\sum_{k=(\mathcal{E}-1)w+1}^{\mathcal{E}w}\sum_{h=0}^{H-1}\varphi(s_h^{k+1}, a_h^{k+1})(\bar{\Lambda}_w^k)^{-1}\varphi(s_h^{k+1}, a_h^{k+1})^\top} \qquad \text{(D.73)} \\
&\le \sum_{\mathcal{E}=1}^{\lfloor \frac{K-1}{w} \rfloor} \sqrt{Hw}\sqrt{\log\left(\frac{\det\left(\Lambda_w^{\mathcal{E}w+1}\right)}{\det\left((\Lambda_w^{(\mathcal{E}-1)w+2})^{(1)}\right)}\right)} \qquad \text{(D.74)} \\
&\le \left\lfloor \frac{K-1}{w} \right\rfloor \sqrt{Hw}\sqrt{\log\left(\frac{\lambda + wH}{\lambda}\right)} \qquad \text{(D.75)} \\
&\le (K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda + wH}{\lambda}\right)}
\end{aligned}$$

Equation (D.73) holds by Cauchy–Schwarz inequality, equation (D.74) holds by the lemma (D.1) and (D.2) of the paper [41], the equation (D.75) holds since $(\Lambda_w^{(\mathcal{E}-1)w+2})^{(1)} \ge \lambda$, $\Lambda_w^{\mathcal{E}w+1} \le \lambda + wH$. □

Lemma 10 (Upper bound of $\Delta_k^p(s,a)$)**.** *For any $(s,a) \in \mathcal{S} \times \mathcal{A}$ and given $\delta \in (0,1)$, the following holds with probability $1-\delta$.*

$$\Delta_{p,k}^1(s,a) \le B_{p,w}^k + (\Lambda_w^k(s,a))^{1/2} \cdot |\mathcal{S}| \cdot \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta\lambda}\right)} + \lambda \Lambda_w^k(s,a)$$

Proof of lemma 10. For any $(s,a) \in \mathcal{S} \times \mathcal{A}$, $\Delta_k^p(s,a)$ is represented as follows.

$$\begin{aligned}
&\Delta_k^p(s,a) \\
&= \|P^{(k+1)}(\cdot|s,a) - \widehat{P}^{(k+1)}(\cdot|s,a)\|_1 \\
&= \|o_{(k+1)}^p(\cdot,s,a) - \widehat{o}_{(k+1)}^p(\cdot,s,a)\|_1 \\
&= \sum_{s' \in \mathcal{S}} \left| \frac{\sum_{t=k-w+1}^k n_t(s',s,a)}{\lambda + \sum_{t=k-w+1}^k n_t(s,a)} - o_{(k+1)}^p(s',s,a) \right| \\
&= \Lambda_w^k(s,a) \sum_{s' \in \mathcal{S}} \left| \sum_{t=k-w+1}^k n_t(s',s,a) - \left(\lambda + \sum_{t=k-w+1}^k n_t(s,a)\right) o_{(k+1)}^p(s',s,a) \right| \\
&\le \Lambda_w^k(s,a) \sum_{s' \in \mathcal{S}} \left(\left| \sum_{t=k-w+1}^k \left(n_t(s',s,a) - n_t(s,a) o_{(k+1)}^p(s',s,a) \right) \right| + \left| \lambda o_{(k+1)}^p(s',s,a) \right| \right) \\
&\le \Lambda_w^k(s,a) \sum_{s' \in \mathcal{S}} \left| \sum_{t=k-w+1}^k \left(n_t(s',s,a) - n_t(s,a) o_{(k+1)}^p(s',s,a) \right) \right| + \lambda \Lambda_w^k(s,a)
\end{aligned} \tag{D.76}$$

Recall that $n_t(s',s,a),\ n_t(s,a)$ is defined as

$$\begin{aligned}
n_t(s',s,a) &= \sum_{h=0}^{H-1} \mathbb{1}\left[(s',s,a) = (s_{h+1}^t, s_h^t, a_h^t)\right] \\
&= \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot \mathbb{1}\left[s' = s_{h+1}^t\right]
\end{aligned} \tag{D.77}$$

$$n_t(s,a) = \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \tag{D.78}$$

where $\mathbb{1}[\cdot]$ is an indicator function. Plugging the equations (D.77),(D.78) into the equations (D.76) gives the following,

$$\begin{aligned}
&\Lambda_w^k(s,a) \sum_{s' \in \mathcal{S}} \left| \sum_{t=k-w+1}^k \left(n_t(s',s,a) - n_t(s,a) o_{(k+1)}^p(s',s,a) \right) \right| \\
&= \Lambda_w^k(s,a) \sum_{s' \in \mathcal{S}} \left| \sum_{t=k-w+1}^k \left(\sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot \mathbb{1}\left[s' = s_{h+1}^t\right] \right.\right. \\
&\left.\left. - \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot o_{(k+1)}^p(s',s,a) \right) \right| \\
&= \Lambda_w^k(s,a) \sum_{s' \in \mathcal{S}} \left| \sum_{t=k-w+1}^k \left(\sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \left(\mathbb{1}\left[s' = s_{h+1}^t\right] - o_{(k+1)}^p(s',s,a) \right) \right) \right| \\
&\le \underbrace{\Lambda_w^k(s,a) \sum_{s' \in \mathbb{S}} \left| \sum_{t=k-w+1}^k \left(\sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \left(\mathbb{1}\left[s' = s_{h+1}^t\right] - o_t^p(s',s,a) \right) \right) \right|}_{(2.1)} \\
&+ \underbrace{\Lambda_w^k(s,a) \sum_{s' \in \mathbb{S}} \left| \sum_{t=k-w+1}^k \left(\sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \left(o_t^p(s',s,a) - o_{(k+1)}^p(s',s,a) \right) \right) \right|}_{(2.2)}
\end{aligned}$$

The term (2.1) can be upperbounded by utilizing the Lemma (34), (43) of the paper [32]. We first denote for any $t \in [K],\ s' \in \mathcal{S}$, the random variable $\eta^t(s') \coloneqq \sum_{h=0}^{H-1} \left(\mathbb{1}\left[s' = s_{h+1}^t\right] - o_t^p(s', s_h^t, a_h^t) \right)$. Now, for given $s' \in \mathcal{S}$ the sequence $\{\eta^\tau(s')\}_{\tau=1}^{\infty}$ is zero-mean and $H/2$-sub Gaussian random variable. From the lemma (43) of the paper [32], set $Y = \lambda \mathbb{I},\ X_t = \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right]$.Then, for given $\delta \in (0,1)$, the following holds with probability $1-\delta$ for any $(s,a) \in \mathcal{S} \times \mathcal{A}$,

$$
\begin{aligned}
&\left| (\Lambda_w^k(s,a))^{1/2} \sum_{t=k-w+1}^{k} \left(\sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot \sum_{h=0}^{H-1} \mathbb{1}[s' = s_{h+1}^t] - o_t^p(s', s, a) \right) \right| \\
&\leq \sqrt{\frac{H^2}{2} \log\left(\frac{(\Lambda_w^k(s,a))^{-1/2} \cdot \lambda^{-1/2}}{\delta/H} \right)} \\
&= \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta} \cdot \frac{1}{(\Lambda_w^k(s,a))^{1/2} \cdot \lambda^{1/2}} \right)} \\
&\leq \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta} \cdot \frac{1}{\lambda} \right)}
\end{aligned} \tag{D.79}
$$

Then the following holds with probability $1-\delta$,

$$
\begin{aligned}
&(2.1) \\
&= (\Lambda_w^k(s,a))^{1/2} \sum_{s' \in \mathcal{S}} \left| (\Lambda_w^k(s,a))^{1/2} \sum_{t=k-w+1}^{k} \left(\sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot \sum_{h=0}^{H-1} \mathbb{1}[s' = s_{h+1}^t] - o_t^p(s', s, a) \right) \right| \\
&\leq (\Lambda_w^k(s,a))^{1/2} \cdot |\mathcal{S}| \cdot \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta\lambda} \right)}
\end{aligned}
$$

The term (2.2) can be bounded as follows,

$$
\begin{aligned}
(2.2) &\leq \Lambda_w^k(s,a) \sum_{s' \in \mathcal{S}} \sum_{t=k-w+1}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \left| o_t^p(s', s, a) - o_{(k+1)}^p(s', s, a) \right| \\
&= \Lambda_w^k(s,a) \sum_{t=k-w+1}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \sum_{s' \in \mathcal{S}} \left| o_t^p(s', s, a) - o_{(k+1)}^p(s', s, a) \right| \\
&= \Lambda_w^k(s,a) \sum_{t=k-w+1}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \left\| o_t^p(\cdot, s, a) - o_{(k+1)}^p(\cdot, s, a) \right\|_1 \\
&\leq \max_{t \in [k-w+1, k]} \left(\left\| o_t^p(\cdot, s, a) - o_{(k+1)}^p(\cdot, s, a) \right\|_1 \right) \cdot \left(\Lambda_w^k(s,a) \sum_{t=k-w+1}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \right) \\
&\leq \max_{t \in [k-w+1, k]} \left(\left\| o_t^p(\cdot, s, a) - o_{(k+1)}^p(\cdot, s, a) \right\|_1 \right) \cdot 1 \\
&\leq B_{p,w}^k(\Delta_\pi)
\end{aligned} \tag{D.80}
$$

Then, combining the equation (D.76), (D.79), (D.80), the $\Delta_{p,k}^1(s,a)$ can be represented as follows,

$$
\Delta_{p,k}^1(s,a) \leq B_{p,w}^k(\Delta_\pi) + (\Lambda_w^k(s,a))^{1/2} \cdot |\mathcal{S}| \cdot \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta\lambda} \right)} + \lambda \Lambda_w^k(s,a)
$$

□

Lemma 11 (Upper bound of $\bar{\Delta}_K^p$)**.** *For given $\delta \in (0,1)$, the following holds with probability $1-\delta$.*

$$
\bar{\Delta}_K^p \leq \left(|\mathcal{S}| \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta\lambda} \right)} + \lambda \right) (K-1) \sqrt{\frac{H}{w}} \sqrt{\log\left(\frac{\lambda + wH}{\lambda} \right)} + wH B_p(\Delta_\pi)
$$

Proof of lemma 11. The total empirical forecasting transition probability model error $\bar{\Delta}_K^p$ can be represented as follows,

$$
\begin{aligned}
\bar{\Delta}_K^p &= \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \bar{\Delta}_{k,h}^p \\
&= \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \Delta_k^p(s_h^{k+1}, a_h^{k+1}) \\
&\leq \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \left((\Lambda_w^k(s_h^{k+1}, a_h^{k+1}))^{1/2} |\mathcal{S}| \sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} \right) \\
&\quad + \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \left(\max_{t\in[k-w+1,k]} \left\| o_t^p(\cdot, s_h^{k+1}, a_h^{k+1}) - o_{(k+1)}^p(\cdot, s_h^{k+1}, a_h^{k+1}) \right\|_1 \right) \\
&\quad + \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \left(\lambda \Lambda_w^k(s_h^{k+1}, a_h^{k+1}) \right) \\
&\leq \left(|\mathcal{S}| \sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda \right) \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \left((\Lambda_w^k(s_h^{k+1}, a_h^{k+1}))^{1/2} \right) \\
&\quad + \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \left(\max_{t\in[k-w+1,k]} \left\| o_t^p(\cdot, s_h^{k+1}, a_h^{k+1}) - o_{(k+1)}^p(\cdot, s_h^{k+1}, a_h^{k+1}) \right\|_1 \right) \\
&\leq \left(|\mathcal{S}| \sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda \right) (K-1) \sqrt{\frac{H}{w}} \sqrt{\log\left(\frac{\lambda + wH}{\lambda}\right)} + wHB_p(\Delta_\pi)
\end{aligned}
$$

□

Prelimnary for proof of theorem 3.

The `W-LSE` involves solving a joint optimization problem (**??**) over $\phi_f^r \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$, $\phi_f^p \in \mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|}$, $q \in \mathbb{R}^N$ to obtain a minimum upper bound of the dynamic regret:

$$
\min_{\phi_f^\diamond, q} \mathcal{L}\left(\phi_f^\diamond, q\ ;\ \square_{1:N}\right) \text{ where } \mathcal{L}\left(\phi_f^\diamond, q\ ;\ \square_{1:N}\right) = \sum_{t=1}^{N} q_t \left(\widehat{\square}_{\phi_f^\diamond}^{k+1} - \square_t \right)^2 + \text{disc}(q) + \frac{1}{wH} \cdot \lambda \|\phi_f^\diamond\|_2 \tag{D.81}
$$

where $\diamond = r$ or p. If $\diamond = r$, then $\square = R(s,a)$ and if $\diamond = p$, then $\square = P(s', s, a)$. $\square_{\phi_f^\diamond}$ means that $\square$ is parameterized by $\phi_f^\diamond$, and $\square_{1:N}$ are observed data of $\square$, and the $\text{disc}(q) \coloneqq \sup_{f\in\mathcal{F}} \left(\mathbb{E}[f(\widehat{\square}^{k+1} \mid \square_{1:N}] - \sum_{t=1}^N q_t \mathbb{E}[\widehat{\square}^t | \square_{1:t-1}] \right)$ measures the non-stationarity of the environment. $\text{disc}(q)$ could be measured and upperbounded by observed data. For example, if $\diamond = r$ and $\square = R$, then ϕ_f^r parameterizes the future reward function $\widehat{R}_{\phi_f^r}^{k+1}$, N is the total number of visits of (s,a) up to episode k, $R_{1:N}(s,a)$ is the set of reward values $\{R_1(s,a), R_1(s,a), .., R_N(s,a)\}$ that the agent has received when visiting (s,a). We demonstrate the modified upper bound of $\mathfrak{R}_\mathcal{I}$ when utilizing `W-LSE`. Before, we define forecasting reward model error as $\Delta_{r,k}^1(s,a) = \left| \left(R^{(k+1)} - \widetilde{R}^{(k+1)} \right)(s,a) \right|$, forecasting transition probability model error as $\Delta_{p,k}^1(s,a) = \left\| \left(P^{(k+1)} - \widehat{P}^{(k+1)} \right)(\cdot \mid s,a) \right\|_1$ where $\widetilde{R}, \widehat{P}^{(k+1)}$ are predicted reward,transition probability from meta-function $\mathcal{I}_{g\circ f}$ (Appendix D.2).

Proof of Theorem 4 . We bring the theorem 7 of the paper [22] to offer the upper bound of the l_2-norm of the reward gap between $R^{(k+1)}(s,a)$ and the $\widetilde{R}^{(k+1)}(s,a)$ as follows. Inspired from the theorem 7 of the paper [22], let us denote $X_{k,h} = (s_h^k, a_h^k) \in \mathcal{S}\times\mathcal{A}$, $Y_{k,h} = R^{(k)}(s_h^k, a_h^k) \in \mathbb{R}$ and assume the environment provides agent an noisy reward $\widehat{Y}_{k,h} = Y_{k,h} + \eta$ with zero-mean gaussian $\eta \sim \mathcal{N}(0, b)$. Define the kernel $\Psi(x) = \varphi(x) \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ where $\varphi(x)$ is the one-hot vector that we have defined on section A.1. Now, let function $r(x) = c^\top \varphi(x)$ where the vector $c \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ is same as the estimated future reward vector $\widetilde{R}^{(k+1)} \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ and $r(x)$ is same as the estimated future reward when $x = (s,a)$, namely $\widetilde{R}^{(k+1)}(s,a)$. Assume, Then,

for given data until episode k, $\mathcal{D}_{data} = \{(X_{1,0}, \widehat{Y}_{1,0}), (X_{1,1}, \widehat{Y}_{1,1}), .., (X_{k,H-1}, \widehat{Y}_{k,H-1})\}$, we denote $\mathcal{D}_{data}^{(s,a)} \coloneqq \{(X_{k,h}, \widehat{Y}_{k,h}) \mid X_{k,h} = (s,a) \text{ where } (X_{k,h}, \widehat{Y}_{k,h}) \in \mathcal{D}_{data}\}$. We relabel $\mathcal{D}_{data}^{(s,a)}$ as $\{\left((s,a), \widehat{Y}_1\right), \left((s,a), \widehat{Y}_2\right), ..., \left((s,a), \widehat{Y}_N\right)\}$ where $N(s,a) = \sum_{t=1}^{k} n_t(s,a)$ is the total number of visitation of (s,a) until k episodes (Definition (D.78)). We deonte N as $N(s,a)$, and $\sum_{t=1}^{N} q_t = 1$. Then for any $(s,a) \in \mathcal{S} \times \mathcal{A}$, the following holds with probability at least $1-\delta$ for all functions $r \in \{x \to c^\top \Psi(x) : \|c\|_2 \le \Lambda\}$,

$$\begin{aligned}\mathbb{E}[\left(r(s,a) - \widehat{Y}_{N+1}\right)^2 | \mathcal{D}_{data}^{(s,a)}] &\le \sum_{t=1}^{N} q_t \left(r(s,a) - \widehat{Y}_t\right)^2 + \text{disc}(q) + \mathcal{O}\left((\log^2 N)\Lambda r \|q\|_2\right) \\ &\le \sum_{t=1}^{N} q_t \left(r(s,a) - \widehat{Y}_t\right)^2 + \text{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{r}\|_2 \end{aligned} \tag{D.82}$$

where we let the function $\mathcal{O}\left((\log^2 N)\Lambda r \|q\|_2\right)$ can be represented as $\frac{1}{wH} \cdot \lambda \|\bar{r}\|_2$ where λ is a constant. Take the expectation over η on both inequailty.

$$\mathbb{E}_\eta\left[\mathbb{E}\left[\left(r(s,a) - \widehat{Y}_{N+1}\right)^2 | \mathcal{D}_{data}^{(s,a)}\right]\right] \le \mathbb{E}_\eta\left[\sum_{t=1}^{N} q_t \left(r(s,a) - \widehat{Y}_t\right)^2 + \text{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{r}\|_2\right]$$

$$\mathbb{E}\left[\left(r(s,a) - \widehat{Y}_{N+1}\right)^2 | \mathcal{D}_{data}^{(s,a)}\right] \le \sum_{t=1}^{N} \mathbb{E}_\eta\left[q_t \left(r(s,a) - \widehat{Y}_t\right)^2\right] + \text{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{r}\|_2$$

The LHS of equation (D.82) can be expressed as follows.

$$\begin{aligned}\mathbb{E}\left[(r(s,a) - \widehat{Y}_{N+1} - \eta)^2\right] &= \mathbb{E}_\eta\left[(r(s,a) - Y_{N+1})^2\right] + \mathbb{E}_\eta\left[\eta^2\right] \\ &= (r(s,a) - Y_{N+1})^2 + \mathbb{E}\left[\eta^2\right] \end{aligned} \tag{D.83}$$

$$\tag{D.84}$$

Also the term$\sum_{t=1}^{N} \mathbb{E}_\eta\left[q_t \left(r(s,a) - \widehat{Y}_t\right)^2\right]$ of the RHS of equation (D.82) is represented as follows.

$$\begin{aligned}\sum_{t=1}^{N} \mathbb{E}_\eta[q_t \left(r(s,a) - \widehat{Y}_t\right)^2] &= \sum_{t=1}^{N} \mathbb{E}_\eta[q_t \left((r(s,a) - Y_t)^2 + \eta^2\right)] \\ &= \sum_{t=1}^{N} \mathbb{E}_\eta\left[q_t \left((r(s,a) - Y_t)^2\right)\right] + \sum_{t=1}^{N} \mathbb{E}_\eta\left[q_t \eta^2\right] \\ &= \sum_{t=1}^{N} q_t \left((r(s,a) - Y_t)^2\right) + \mathbb{E}_\eta\left[\eta^2\right] \end{aligned}$$

Then the term $\mathbb{E}_\eta[\eta^2]$ removed by both side, then the following holds,

$$(r(s,a) - Y_{N+1})^2 \le \sum_{t=1}^{N} q_t \left((r(s,a) - Y_t)^2\right) + \text{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{r}\|_2 \tag{D.85}$$

Recall the definition of $r(s,a) = \widetilde{R}^{(k+1)}(s,a)$, $Y_t = R_t(s,a)$. Since t matches with one of $(k,h) \in [K] \times [H]$ pairs, we can rewrite $\sum_{t=1}^{N} q_t \left(r(s,a) - \widehat{Y}_t\right)^2 = \sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \left(r(s,a) - Y_{(k,h)}\right)^2 = \sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \left(\widetilde{R}^{(k+1)}(s,a) - R_h^{k'}(s,a)\right)^2 = \sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \left(\widetilde{R}^{(k+1)}(s,a) - R^{k'}(s,a)\right)^2$ where if (s,a) is not visited at step h of episode k, then corresponding $q_{(k',h)}$ is zero. Then, the following holds,

$$\begin{aligned}\Delta_{r,k}^1(s,a) &\le \sqrt{\min_{q,\bar{r}} \left(\sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \left(\widetilde{R}^{(k+1)}(s,a) - R^{k'}(s,a)\right)^2 + \text{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{r}\|_2\right)} \\ &\le \sqrt{\min_{q,\bar{r}} \left(\left(\max_{1 \le k' \le k} \left(\widetilde{R}^{(k+1)}(s,a) - R^{k'}(s,a)\right)\right)^2 \left(\sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)}\right) + \text{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{r}\|_2\right)} \end{aligned}$$

Similar process for $\Delta^1_{p,k}$ provides the following inequality. For any $(s', s, a) \in \mathcal{S} \times \mathcal{S} \times \mathcal{A}$,

$$\begin{aligned}
&\left|P^{(k+1)}(s' \mid s, a) - \widehat{P}^{(k+1)}(s' \mid s, a)\right| \\
&\leq \sqrt{\min_{q,\bar{p}} \left(\sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \left(\widehat{P}^{(k+1)}(s'|s,a) - P^{k'}(s'|s,a) \right)^2 + \mathrm{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{p}\|_2 \right)}
\end{aligned}$$

holds. Then, recall the definition of $\Delta^1_{p,k}(s,a)$.

$$\begin{aligned}
\Delta^1_{p,k}(s,a) &\leq \sum_{s' \in \mathcal{S}} \sqrt{\min_{q,\bar{p}} \left(\sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \left(\widehat{P}^{(k+1)}(s'|s,a) - P^{k'}(s'|s,a) \right)^2 + \mathrm{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{p}\|_2 \right)} \\
&\leq |\mathcal{S}| \sqrt{\min_{q,\bar{p}} \left(\sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \left\| \widehat{P}^{(k+1)}(\cdot|s,a) - P^{k'}(\cdot|s,a) \right\|_\infty^2 + \mathrm{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{p}\|_2 \right)}
\end{aligned}$$

Now, recall the Corollary 5,6 and definition of $\mathfrak{R}_{\mathcal{I}}$. Fix (s,a), then $\mathfrak{R}_{\mathcal{I}}(s,a)$

$$\begin{aligned}
\mathfrak{R}_{\mathcal{I}} =& \frac{1}{1-\gamma} \sum_{k=1}^{K-1} \bar{\iota}_\infty^{k+1} + \sum_{k=1}^{K-1} \sum_{h=0}^{H-1} -\iota_h^{k+1} + C_p \sqrt{K-1} \\
\leq& \frac{1}{1-\gamma} \sum_{k=1}^{K-1} \left(\Delta^1_{r,k}(s,a) + \Delta^1_{p,k}(s,a) \frac{\gamma \hat{r}_{max}}{1-\gamma} - 2\Gamma^k_w(s,a) \right) \\
&+ \sum_{k=1}^{K-1} \sum_{h=0}^{H-1} \left(\Delta^1_{r,k}(s,a) + \Delta^p_k(s,a) \frac{\gamma \hat{r}_{max}}{1-\gamma} + 2\Gamma^k_w(s,a) \right) \\
&+ C_p \sqrt{K-1} \\
\leq& \frac{1}{1-\gamma} \sum_{k=1}^{K-1} \left(\Delta^1_{r,k}(s,a) + \Delta^1_{p,k}(s,a) \frac{\gamma}{1-\gamma} \left(\tilde{r}_{\max} + \max(2\Gamma^k_w(s,a)) \right) - 2\Gamma^k_w(s,a) \right) \\
&+ H \sum_{k=1}^{K-1} \left(\Delta^1_{r,k}(s,a) + \Delta^p_k(s,a) \frac{\gamma}{1-\gamma} \left(\tilde{r}_{\max} + \max(2\Gamma^k_w(s,a)) \right) + 2\Gamma^k_w(s,a) \right) \\
&+ C_p \sqrt{K-1} \\
\leq& \sum_{k=1}^{K-1} \Bigg(\underbrace{\left(\left(\frac{1}{1-\gamma} + H \right) \Delta^1_{r,k}(s,a) + \frac{\gamma \tilde{r}_{\max}}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \Delta^1_{p,k}(s,a) \right)}_{①} + \\
&+ \frac{\gamma}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \max(2\Gamma^k_w(s,a)) \Delta^1_{p,k}(s,a) \Bigg) \\
&+ \sum_{k=1}^{K-1} 2 \left(-\frac{1}{1-\gamma} + H \right) \Gamma^k_w(s,a) \\
&+ C_p \sqrt{K-1}
\end{aligned}$$

Now we let the term ① as $2(\frac{1}{1-\gamma} + H)\Gamma^k_w(s,a)$, that is if we redefine the exploration bonus term as follow,

$$\Gamma^k_w(s,a) = \frac{1}{2} \Delta^1_{r,k}(s,a) + \frac{\gamma \tilde{r}_{\max}}{2(1-\gamma)} \Delta^1_{p,k}(s,a)$$

Also, note that $\Delta_k^p(s,a) = \sum_{s' \in \mathcal{S}} \left| \left(\widehat{P}^{(k+1)} - P^{(k+1)} \right)(s'|s,a) \right| \leq |\mathcal{S}|$. Then the following holds,

$$
\begin{aligned}
\mathfrak{R}_{\mathcal{I}} &\leq \sum_{k=1}^{K-1} \left(4H\Gamma_w^k(s,a) + \frac{2\gamma}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \max(\Gamma_w^k(s,a)) \, |\mathcal{S}| \right) \\
&\leq \sum_{k=1}^{K-1} \left(4H + \frac{2\gamma}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \right) \max(\Gamma_w^k(s,a)) \\
&= \left(4H + \frac{2\gamma|\mathcal{S}|}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \right) \sum_{k=1}^{K-1} \max(\Gamma_w^k(s,a)) \\
&\leq \left(4H + \frac{2\gamma|\mathcal{S}|}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \right) \sum_{k=1}^{K-1} \left(\frac{1}{2} \max(\Delta_{r,k}^1(s,a)) + \frac{\gamma \tilde{r}_{\max}}{2(1-\gamma)} \max(\Delta_{p,k}^1(s,a)) \right) \\
&= \left(4H + \frac{2\gamma|\mathcal{S}|}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \right) \sum_{k=1}^{K-1} \left(\frac{1}{2} \Delta_{r,k}^1(s,a) + \frac{\gamma \tilde{r}_{\max}}{2(1-\gamma)} \Delta_{p,k}^1(s,a) \right) \\
&= \left(4H + \frac{2\gamma|\mathcal{S}|}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \right) \left(\frac{1}{2} \sum_{k=1}^{K-1} \Delta_{r,k}^1(s,a) + \frac{\gamma \tilde{r}_{\max}}{2(1-\gamma)} \sum_{k=1}^{K-1} \Delta_{p,k}^1(s,a) \right)
\end{aligned}
$$

Note that above upper bound for $\mathfrak{R}_{\mathcal{I}}$ holds for following given upper bound condition of $\Delta_{r,k}^1(s,a), \Delta_{p,k}^1(s,a)$,

$$
\Delta_{r,k}^1(s,a) \leq \sqrt{\min_{q,\bar{r}} \left(\sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \left(\widetilde{R}^{(k+1)}(s,a) - R^{k'}(s,a) \right)^2 + \mathrm{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{r}\|_2 \right)}
$$

$$
\Delta_{p,k}^1(s,a) \leq \sum_{s' \in \mathcal{S}} \sqrt{\min_{q,\bar{p}} \left(\sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \left(\widehat{P}^{(k+1)}(s'|s,a) - P^{k'}(s'|s,a) \right)^2 + \mathrm{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{p}\|_2 \right)}
$$

□

Proof of Remark 1. The proof starts from the equation (D.85). Let us denote r^{sw}, q^{sw} as follows.

$$
q_t^{sw} = \begin{cases} \frac{1}{wH} & \text{if } t \in (k-w, k] \\ 0 & \text{otherwise} \end{cases},
$$

$$
r^{sw} = \arg\min_{\bar{r}} \left(\lambda \|\bar{r}\|^2 + \sum_{t=1}^{N} \left(r(s,a) - \widehat{Y}_t \right)^2 \right) \tag{D.86}
$$

where r_{sw} is the same reward estimation with equation D.7. Then the minimum over $\bar{r}, q$ of the equation (D.82) yields the following inequality,

$$
\begin{aligned}
&\min_{\bar{r},q} \left(\sum_{t=1}^{N} q_t \left(r(s,a) - \widehat{Y}_t \right)^2 + \mathrm{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{r}\|_2 \right) && \text{(D.87)} \\
&\leq \min_{\bar{r}} \left(\sum_{t=1}^{N} q_t^{sw} \left(r(s,a) - \widehat{Y}_t \right)^2 + \mathrm{disc}(q^{sw}) + \frac{1}{Hw} \cdot \lambda \|\bar{r}\|_2 \right) \\
&\leq \frac{1}{Hw} \underbrace{\min_{\bar{r}} \left(\sum_{t=1}^{N} (Hw) \cdot q_t^{sw} \left(r(s,a) - \widehat{Y}_t \right)^2 + \lambda \|\bar{r}\|_2 \right)}_{①} + \mathrm{disc}(q_{sw}) && \text{(D.88)}
\end{aligned}
$$

The term ① is the optimization problem of equation (D.86) where its minimizer is r^{sw}. The inspection through equation (D.87) ~ (D.88) provides that the optimal $q^*, \bar{r}^*$, the minimizer of equation (D.87), can provide smaller value than q^{sw}, r^{sw}. Since the RHS of the equation (D.85) is same as the equation (D.87), $q^*, \bar{r}^*$ provide a tighter upper bound of the LHS term of equation (D.82) than q^{sw}, r^{sw}. Therefore, the equation (D.83) provides that the optimal $q^*, \bar{r}^*$ gives a tighter upper bound of the term $\Delta_{r,k}^1$ using than $q^{sw}, \bar{r}^{sw}$. Repeat similar analysis for the upper bound of the $\Delta_{p,k}^1$. Then, by Corollary 5,6, the tighter upper bound of $\Delta_{r,k}^1(s,a), \Delta_{p,k}^1(s,a)$ provides the smaller upper bound of $-\iota_H^K, \bar{\iota}_\infty^K$ and leads to tighter upper bound of $\mathfrak{R}_{\mathcal{I}}$.

□

E Experimental design and results

E.1 Environment setting details

Reward function design.

All three environments share the same reward function structure and have an identical goal. The reward function R consists of three parts $\mathcal{R} = \mathcal{R}_h + \mathcal{R}_f - \mathcal{R}_c$ where $\mathcal{R}_h$ is the healthy reward, $\mathcal{R}_f = k_f(x_{t+1} - x_t)/\Delta t, k_f > 0$ is the forward reward, and R_c is the control cost. Agents have a goal to run faster in the $+x$ direction, so the faster they run, the higher the forward reward $\mathcal{R}_f$ they receive. We modified the environment to make the agent's desired directions change as the episode goes by. To be specific, we designed the forward reward R_f to change as episodes progress as $\mathcal{R}_f^k = o_k \cdot k_f(x_{t+1} - x_t)/\Delta t$ where o_k is the sine function $a\sin(wbk)$ where $a, b > 0$ are constant. Positive o_k causes the agent to desire the forward $+x$ direction as an optimal policy, and negative o_k causes it to desire the backward $-x$ direction. We generated different speeds of non-stationary by changing the frequency variable w through [1,2,3,4,5].

Non-stationary variable o_k generator.

1. Sine function : The non-stationary variable o_k is designed as $o_k = \sin(2\pi wk/37)$ where w is the integer speed of the environment change and k is the episode. We changed w among [1,2,3,4,5]. We divided $2\pi wk$ by 37, the prime number, to make sure that the environment has various non-stationary modes and to avoid certain non-stationary variables appearing frequently.
2. Real data : we brought the stroke price data to model the non-stationary real data. We use real world data that the function that generates the non-staionary variable is not known.

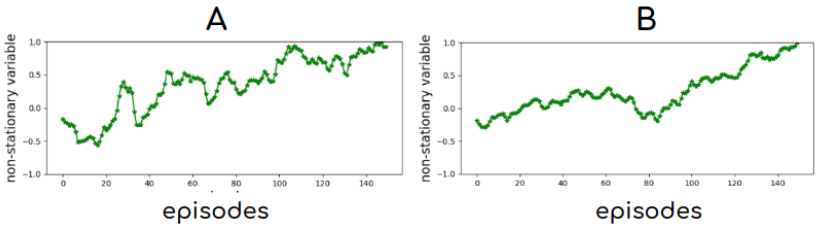

Figure 4: Nonstationary variable from real dataA,B

Non-stationary variable o_k generator (ablation study). We provide for o_k that we used in our expierments, $B(G)$ satisfies the property of the time-elapsing variation budget that $B(G)$ increases as G increases as follows. For the ablation study, we generate $o_k = \sin(2\pi \cdot G \cdot k/37)$ where $G = [38, 76, 114, 152, 190]$. We estimated $B(G)$ as $\sum_{k=1}^{150} |o_{k+1} - o_k|$

	$\Delta_t = 1/G = 38$	$\Delta_t = 2/G = 76$	$\Delta_t = 3/G = 114$	$\Delta_t = 4/G = 152$	$\Delta_t = 5/G = 190$
$B(G)$	15.98	31.85	47.49	62.79	77.64

E.2 Hyperparameters and implementation details

Training Details.

For the ARIMA model that serves as a forecatser f, we used auto_arima function of pmdarima python package to find the optimal p, q, d. To compare the result between `ProST-G` and MBPO, we both trained the MBPO and `ProST-G` with initial learning rate $lr = 3e-4$ with the decaying parameter 0.999. For `ProST-G`, We added uniform noise $\eta \sim \text{Unif}([-b, b])$ in observed non-stationary variable o^k to generate noisy non-stationary variable $\hat{o}_k = o_k + \eta$ with different noise bound $b \in [0.01, 0.03, 0.05]$).

To compare the result between `ProST-G` with ProOLS,ONPG,FTML, we trained these three baselines with eight different initial learning rate $lr = \{1e-3, 3e-3, 5e-3, 7e-3, 1e-2, 3e-2, 5e-2, 7e-2\}$.

Hyper parameters.

Letter	hyper parameters	Swimmer-v2	Half cheetah-v2	Hopper-v2
K	episodes	100	150	150
H	environment steps per episodes	100		
G	policy updates per epochs	50		
$\widehat{H}$	model rollout length	1 → 15 over episodes 20 →150		
N	iteration of policy update and policy evaluation	1		
M	model rollout batch size (D_{syn})	1e5		
τ	entropy regularization parameter	0.2		
γ	reward discounting factor	0.99		

Note that $\widehat{H}$ increases linearly within a certain range as episode goes by. We denote $h_{min} \rightarrow h_{max}$ over episodes $k_{min} \rightarrow k_{max}$ as $\widehat{H}(k) = \min(\max(h_{min} + (k - k_{min})/(k_{max} - k_{min}) \cdot (h_{max} - h_{min}), h_{min}), h_{max})$.

E.3 Full results

E.3.1 Non-stationarity : sin wave

(1) Swimmer-v2

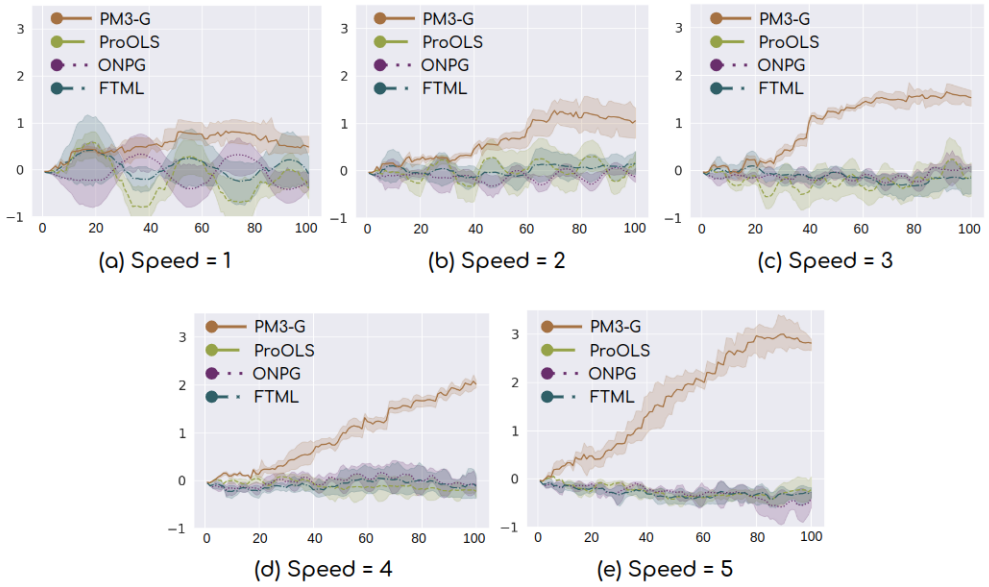

Figure 5: (a) ~(e) the average rewards of `ProST-G`, and the three baselines : ProOLS, ONPG, FTML for 5 different speeds (x-axis is episode). The shaded area of `ProST-G` is 95 % confidence area among 3 different noise bounds (0.01, 0.03, 0.05), and the shaded ares of three baselines are the 95 % confidence area among 8 different learning rates (0.001, 0.003, 0.005, 0.007, 0.01, 0.03, 0.05, 0.07).

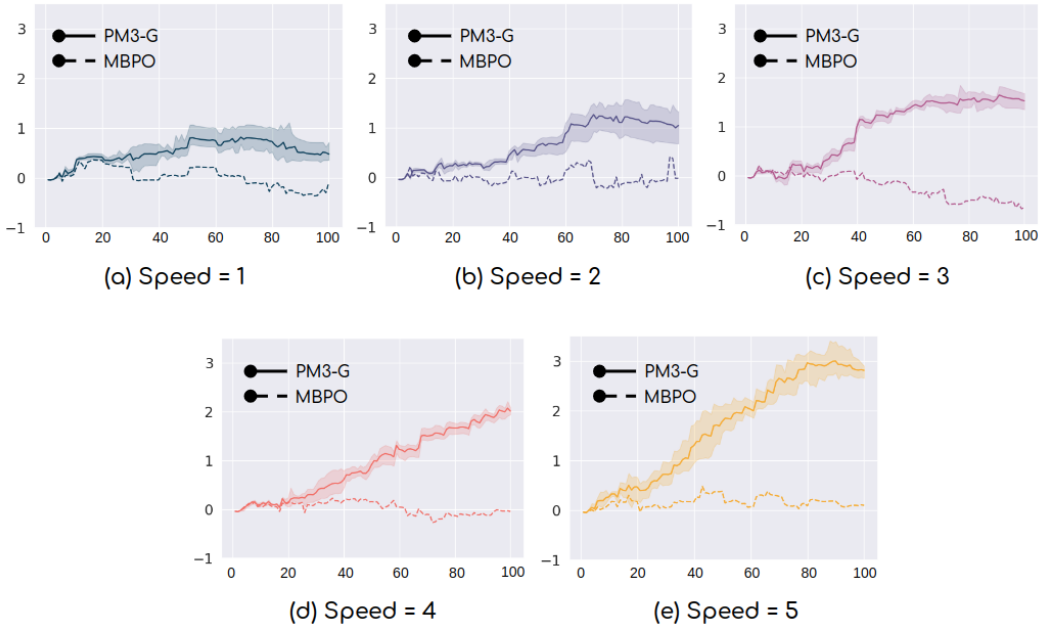

Figure 6: (a) ~(e) the average rewards of `ProST-G` and MBPO. The shaded area of `ProST-G` is 95 % confidence area among 3 different noise bound (0.01, 0.03, 0.05).

(2) Halfcheetah-v2

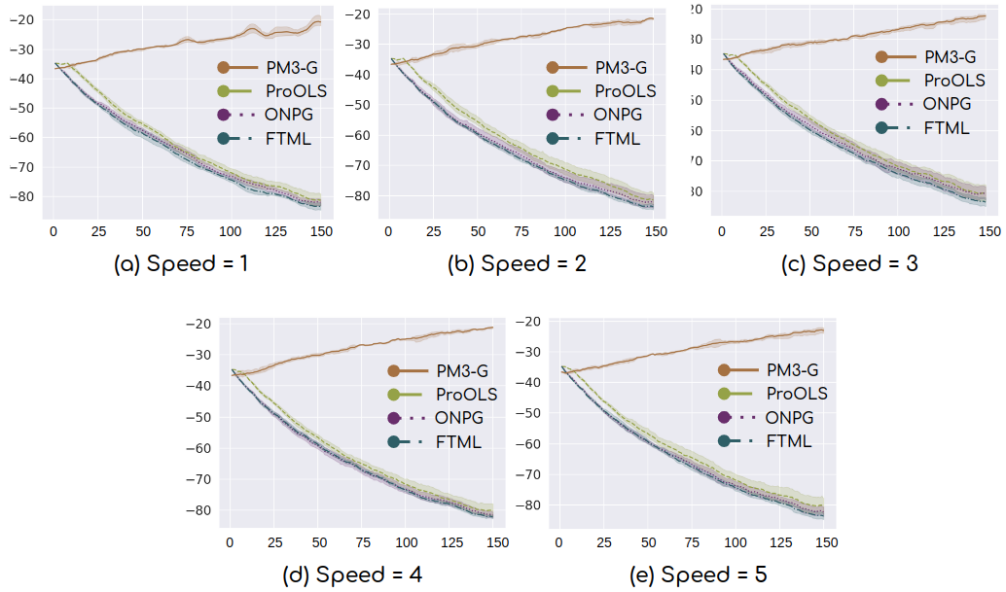

Figure 7: (a) ~(e) the average rewards of `ProST-G`, and the three baselines : ProOLS, ONPG, FTML for 5 different speeds (x-axis is episode). The shaded area of `ProST-G` is 95 % confidence area among 3 different noise bounds (0.01, 0.03, 0.05), and the shaded ares of three baselines are the 95 % confidence area among 8 different learning rates (0.001, 0.003, 0.005, 0.007, 0.01, 0.03, 0.05, 0.07).

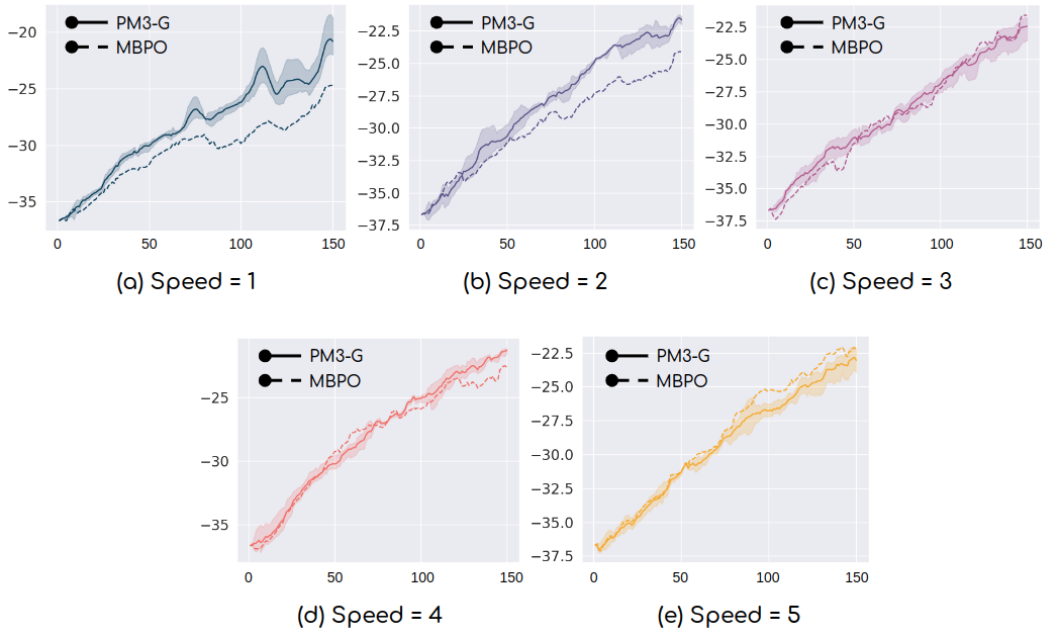

Figure 8: (a) ~(e) the average rewards of `ProST-G` and MBPO (x-axis is episode). The shaded area of `ProST-G` is 95 % confidence area among 3 different noise bound (0.01, 0.03, 0.05).

(3) Hopper-v2

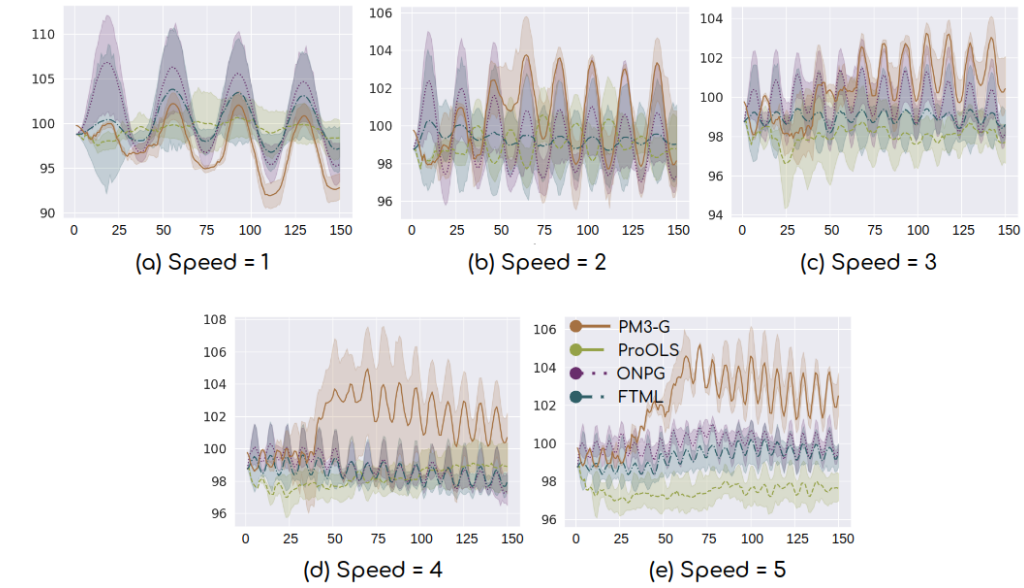

Figure 9: (a) ~(e) the average rewards of `ProST-G`, and the three baselines : ProOLS, ONPG, FTML for 5 different speeds (x-axis is episode). The shaded area of `ProST-G` is 95 % confidence area among 3 different noise bound (0.01, 0.03, 0.05), and the shaded ares of three baselines are the 95 % confidence area among 8 different learning rates (0.001, 0.003, 0.005, 0.007, 0.01, 0.03, 0.05, 0.07).

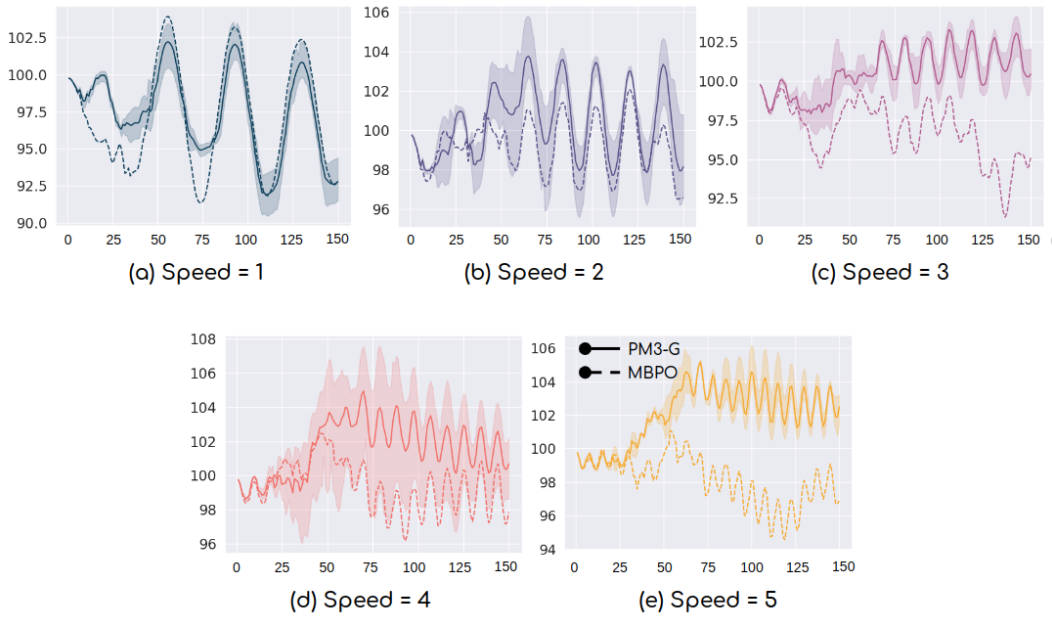

Figure 10: (a) ~(e) the average rewards of `ProST-G` and MBPO (x-axis is episode). The shaded area of `ProST-G` is 95 % confidence area among 3 different noise bound (0.01, 0.03, 0.05).

E.3.2 Non-stationarity : real data

The shaded area of ProST-G is 95 % confidence area among 3 different noise bounds (0.01, 0.03, 0.05), and the shaded ares of three baselines are the 95 % confidence area among 8 different learning rates (0.001, 0.003, 0.005, 0.007, 0.01, 0.03, 0.05, 0.07).

(1) Swimmer-v2

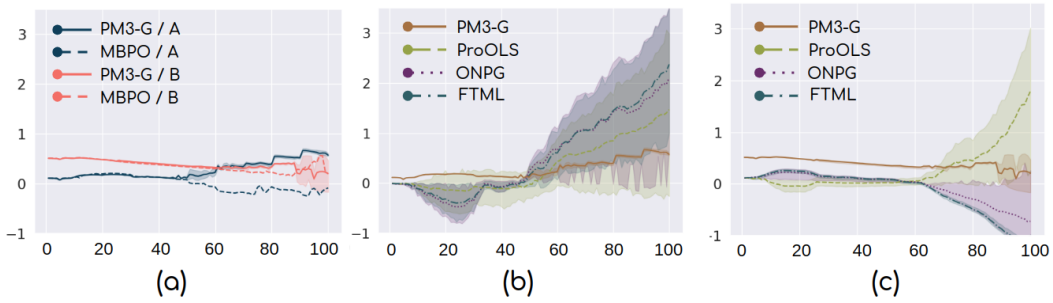

Figure 11: (a) average reward with ProST-G and MBPO on real data A,B (x-axis is episode). (b) average reward with ProST-G and three baselines on realdata A. (c) average reward with ProST-G and three baselines on realdata B.

(2) Halfcheetah-v2

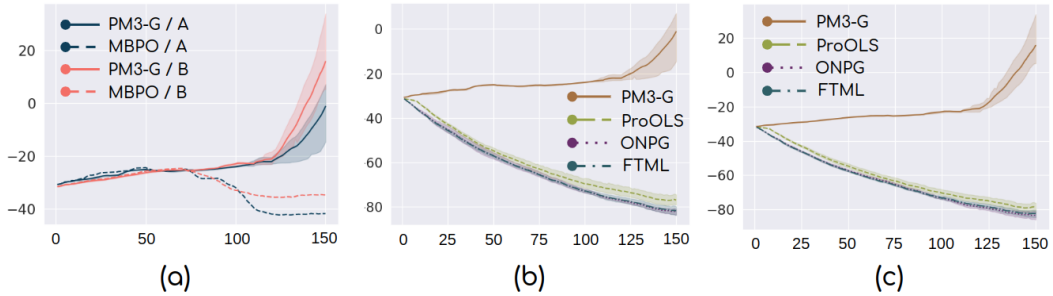

Figure 12: (a) average reward with ProST-G and MBPO on real data A,B (x-axis is episode). (b) average reward with ProST-G and three baselines on realdata A. (c) average reward with ProST-G and three baselines on realdata B.

(3) Hopper-v2

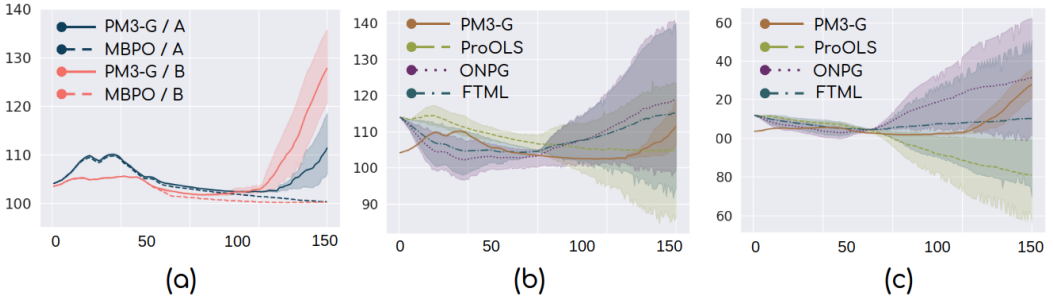

Figure 13: (a) average reward with ProST-G and MBPO on real data A,B (x-axis is episode). (b) average reward with ProST-G and three baselines on realdata A. (c) average reward with ProST-G and three baselines on realdata B.

F Meta algorithm design

F.1 Meta-algorithm $\mathcal{I}_{g\circ f}[A]$

Algorithm 1: Meta-algorithm $\mathcal{I}_{g\circ f}[A]$

Set : $k_f = 1$
Init : policy π^0, forecaster $f_{\phi_f^0}$, model estimator $g_{\phi_g^0}$, two dataset $\mathcal{D}_{env}, \mathcal{D}_{syn}$
for *episode* k **do**
  Execute the agent with π^k in a environment $\mathcal{M}_k$ and add a trajectory to $\mathcal{D}_{env}$.
  /* Meta function $\mathcal{I}_{g\circ f}$ */
  /* (1) Observe and forecast : */
  Observe a noisy non-stationary variable $\hat{o}_k$
  Update f_{ϕ_f}, g_{ϕ_g} using $\mathcal{D}_{env}$ and $\hat{o}_{k-(w-1):k}$.
  Use $f_{\phi_f^k}, g_{\phi_g^k}$ to predict the future $\widehat{\mathcal{P}}^{k+1}$, $\widehat{\mathcal{R}}^{k+1}$ and construct future MDP $\widehat{\mathcal{M}}_{k+1}$
  /* Baseline A */
  /* (2) Optimize : */
  Roll out synthetic trajectories in the $\widehat{\mathcal{M}}_{k+1}$ and add them to $\mathcal{D}_{syn}$
  Use $\mathcal{D}_{syn}$to evaluate and update π^k to $\widehat{\pi}^{k+1}$
end for

F.2 `ProST-T` algorithm

Algorithm 2: Forecasted tabular model based policy optimization (`ProST-T`)

Set : $k_f = 1$
Init : policy π^k , forecaster $f_{\phi_f^k}$, tabular reward model g_k^R, tabular transition probability model g_k^P, forecasted state-action value $\widehat{Q}^{\cdot,k+1}$, empty dataset $\mathcal{D}_{env}, \mathcal{D}_{syn}$
Explore w episodes and add $(\tau^{-k}, \hat{o}_{-k})$ to D_{env} where $k \in [w]$ before starts
for *episodes* $k = 1, .., K$ **do**
  Rollout a trajectory τ_k using $\pi^k$and $\mathcal{D}_{env} = \mathcal{D}_{env} \cup \{\tau_k\}$
  Observe a noisy non-stationary variable $\hat{o}_k$
  /* Meta function $\mathcal{I}_{g\circ f}$: (1) update f, g */
  Update f_{ϕ_f} : $\phi_f^k \leftarrow \arg\min_\phi \mathcal{L}_f(\hat{o}_{k-(w-1):k}; \phi)$
  Update $g_k^P(s', s, a, o)$
  Update $g_k^R(s, a, o)$
  /* Meta function $\mathcal{I}_{g\circ f}$: (2) predict $\widehat{\mathcal{P}}^{k+1}, \widehat{\mathcal{R}}^{k+1}$ */
  Forecast 1 episode ahead non-stationary variable : $\hat{o}_{k+1} = f_{\phi_f^k}(\hat{o}_{k-(w-1):k})$
  Forecast transition probability function: $\widehat{g}_{k+1}^P = g_k^P(\cdot, \hat{o}_{k+1})$
  Forecast reward function: $\widehat{g}_{k+1}^R = g_k^R(\cdot, \hat{o}_{k+1})$
  Reset $\mathcal{D}_{syn}$ to empty.
  /* Baseline A : NPG with entropy regularzation */
  Set $\widehat{\pi}^{(0)} \leftarrow \pi^k$
  for $g = 0, .., G-1$ **do**
    Evaluate $Q_\tau^{\widehat{\pi}^{(g)}}$ using the rollouts from the future model $\widehat{g}_{k+1}^P, \widehat{g}_{k+1}^R$
    Update $\widehat{\pi}$: $\widehat{\pi}^{(g+1)} \leftarrow 1/Z^{(t)} \cdot \left(\widehat{\pi}^{(g)}\right)^{1-\frac{\eta\tau}{1-\gamma}} \exp\left((\eta \widehat{Q}_\tau^{\widehat{\pi}^{(g)}})/(1-\gamma)\right)$
    where $Z^{(t)} = \sum_{a\in\mathcal{A}} \left(\widehat{\pi}^{(g)}\right)^{1-\frac{\eta\tau}{1-\gamma}} \exp\left((\eta \widehat{Q}_\tau^{\widehat{\pi}^{(g)}})/(1-\gamma)\right)$
  end for
  set $\pi^{k+1} \leftarrow \widehat{\pi}^{(G)}$
end for

F.3 ProST-G algorithm

(1) Forecaster f. We adopt the ARIMA model to forecast $\hat{o}_{k+1}$ from the noisy observed $\widehat{o}_{k-(w-1):k}$. ARIMA model is one of the most general class of models for forecasting a time series which can be made to be stationary by taking a difference among the data. For given time series data X_t, we define ARIMA(p,d,q) as given by $X_t - \alpha_1 X_{t-1} - ... - \alpha_p X_{t-p} = \epsilon_t + \theta_1 \epsilon_{t-1} + ... + \theta_q \epsilon_{t-q}$ where α_i are the parameters of the autoregressive part of the model, the θ_i are the parameters of the moving average part and ϵ_t are the error terms that take d times difference between X_ts, which we assume to be independent and follow a normal distribution with zero mean.

(2) Model predictor g. We use a bootstrap ensemble of dynamic models $\{g^1_{\phi_g}, g^1_{\phi_g}, ..., g^M_{\phi_g}\}$. Each ensemble model is a probabilistic neural network whose output is parameterized by the mean vector μ and the diagonal vector of the standard deviation Diag(Σ) of a Gaussian distribution , $g^i_{\phi_g}(s_{h+1}, r_h | s_h, a_h, \widehat{o}_{k+1}) = \mathcal{N}(\mu^i_{\phi_g}(s_h, a_h), \Sigma^i_{\phi_g}(s_h, a_h))$. To efficiently handle uncertainty due to the non-stationary environment, we designed each neural network to be a probabilistic model to capture the aleatoric uncertainty, the noise of the output, and learn multiple models as bootstrap ensemble to handle the epistemic uncertainty, the uncertainty in the model parameters. Then we predict s_{h+1}, r_h from a model uniformly chosen from its ensemble randomly that admits different transitions along a single model rollout to be sampled from different dynamics modes.

(3) Baseline algorithm A. We adopt soft-actor critic (SAC) as our policy optimization algorithm. SAC alternates the policy evaluation step and the policy optimization step. For a given policy $\widehat{\pi}$, it estimates the forecasted $\widehat{Q}^{\widehat{\pi},k+1}$ value using Bellman backup operator and optimizes the policy that minimizes the expected KL-divergence between π and the exponential of the difference $\widehat{Q}^{\widehat{\pi},k+1} - \widehat{V}^{\widehat{\pi},k+1}$: $\mathbb{E}_{s\sim D_{syn}}[D_{KL}(\widehat{\pi} \| \exp(\widehat{Q}^{\widehat{\pi},k+1} - \widehat{V}^{\widehat{\pi},k+1})]$

Algorithm 3: Forecasted-Model Based Policy Optimization ((ProST-G)

Set : $k_f = 1$
Init : policy π^k, forecaster $f_{\phi^k_f}$, model estimator $g_{\phi^k_g}$, two dataset $\mathcal{D}_{env}, \mathcal{D}_{syn}$
Explore w episodes and add $(\tau^{-k}, \hat{o}_{-k})$ to D_{env} where $k \in [w]$ before starts
for *episodes* $k = 1, .., K$ **do**
    Execute the agent with π^k in a environment $\mathcal{M}_k$ and add a trajectory to $\mathcal{D}_{env}$.
    /* Meta function $\mathcal{I}_{g\circ f}$: (1) update f, g */
    Observe a noisy non-stationary variable $\hat{o}_k$
    Optimize $f_{\phi^k_f}$ on $\hat{o}_{k-(w-1):k}$
    Optimize $g_{\phi^k_g}$ on $\mathcal{D}_{env}$
    /* Meta function $\mathcal{I}_{g\circ f}$: (2) predict f, g */
    Forecast $\hat{o}_{k+1} = f_{\phi^k_f}(\hat{o}_{k-(w-1):k})$
    Forecast model : $\widehat{g}_{k+1} = g_{\phi^k_g}(\cdot, \hat{o}_{k+1})$
    Reset $\mathcal{D}_{syn}$ to empty.
    /* Baseline A : SAC */
    Set $\widehat{\pi}^{k+1} \leftarrow \pi^k$
    for *epochs* $n = 1, ..., N$ **do**
        for *model rollouts* $m = 1, .., M$ **do**
            Sample $\hat{s}^m_0$ uniformly from D_{env}.
            Perform a $\widehat{H}$-step model rollout using $\hat{a}^m_h = \widehat{\pi}^{k+1}(\hat{s}^m_h)$, $\hat{s}^m_{h+1} = \widehat{g}_{k+1}(\hat{s}^m_h, \hat{a}^m_h)$ and add a rollout to $\mathcal{D}_{syn}$.
        end for
        for *updates* $g = 1, .., G$ **do**
            Evaluate and update forecasted policy $\widehat{\pi}^{k+1}$ on $\mathcal{D}_{syn}$
        end for
    end for
    Set $\pi_{k+1} \leftarrow \widehat{\pi}^{k+1}$
end for

G Experiment Platforms and Licenses

G.1 Platforms

All experiments are done on 12 Intel Xeon CPU E5-2690 v4 and 2 Tesla V100 GPUs.

G.2 Licenses

We have used the following libraries/ repos for our python codes:

- Pytorch (BSD 3-Clause "New" or "Revised" License).
- OpenAI Gym (MIT License).
- Numpy (BSD 3-Clause "New" or "Revised" License).
- Official codes distributed from the paper [7]: to compare the four baselines.
- Official codes distributed from the paper [24]: to build `PMT-G`.