Tempo Adaptation 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, Berkeley, CA 94709
[2]Virginia Tech, Blacksburg, VA 24061
{hyunin,yuhao_ding,jongmin.lee,lavaei,sojoudi}@berkeley.edu
jinming@vt.edu

Abstract

We first raise and tackle a "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 ($\mathfrak{t}$) rather than episode progress (k), where wall-clock time signifies the actual elapsed time within the fixed duration $\mathfrak{t} \in [0, T]$. In existing works, at episode k, the agent rolls 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 $\mathfrak{t}_k$ allocates $\Delta\mathfrak{t}$ for trajectory generation and training, subsequently moves to the next episode at $\mathfrak{t}_{k+1} = \mathfrak{t}_k + \Delta\mathfrak{t}$. Despite a fixed total number of episodes (K), the agent accumulates different trajectories influenced by the choice of *interaction times* $(\mathfrak{t}_1, \mathfrak{t}_2, ..., \mathfrak{t}_K)$, significantly impacting the suboptimality gap of the policy. We propose a Proactively Synchronizing Tempo (`ProST`) framework that computes a suboptimal sequence $\{\mathfrak{t}_1, \mathfrak{t}_2, ..., \mathfrak{t}_K\} (= \{\mathfrak{t}\}_{1:K})$ by minimizing an upper bound on its performance measure, i.e., the dynamic regret. Our main contribution is that we show that a suboptimal $\{\mathfrak{t}\}_{1:K}$ trades-off between the policy training time (agent tempo) and how fast the environment changes (environment tempo). Theoretically, this work develops a suboptimal $\{\mathfrak{t}\}_{1:K}$ 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 suboptimal $\{\mathfrak{t}\}_{1:K}$ than the existing methods.

1 Introduction

The prevailing reinforcement learning (RL) 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 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

* Corresponding authors. This work was supported by grants from ARO, ONR, AFOSR, NSF, and the UC Noyce Initiative.

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

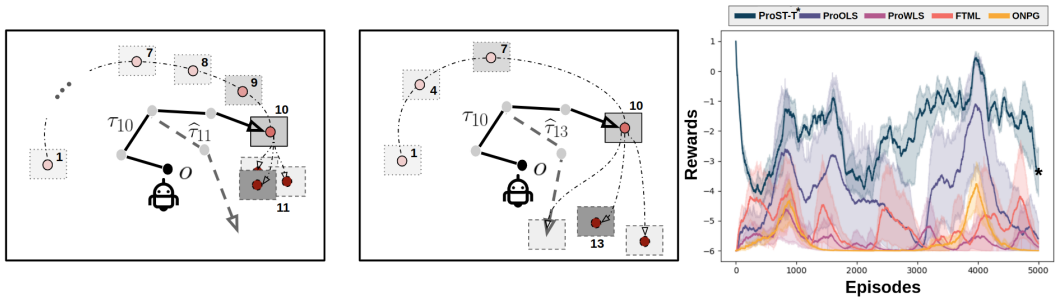

Figure 1: (a) 2D goal reacher in a time-desynchronized environment for one policy update, where the agent learns an inaccurate policy on an accurate model; (b) For three policy updates, the agent learns a near-optimal policy on an inaccurate model; (c) Rewards per episode in 2D goal reacher with four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

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 to execute the episode k, the duration of policy training between the episodes k and $k+1$, and the transition time to the episode $k+1$. The issue stems from the premise that the 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 a real-world environment, thereby the 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 collected trajectories, 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 suboptimality gap of the policy. In the context of a time-desynchronized environment, achieving an optimal policy requires addressing a previously unexplored question: the determination of the *optimal time sequence* $\{\mathrm{t}_1, \mathrm{t}_2, .., .\mathrm{t}_K\} (= \{\mathrm{t}\}_{1:K})$ at which the agent should interact with the environment.

We elucidate the significance of the aforementioned research question through an example. Consider a robot with the goal of reaching inside a gray-shaded non-fixed target box, known as the goal reacher (Appendix A.1). Note that the reward changes as the position of the box changes over time (Figure 1-(a)). We begin by considering a scenario in which the 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, and 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 a faster adaptation to achieve 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 the 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 task of determining both the total number of interactions (denoted as the total episode K) and the specific time instances 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, that is, $\{\mathrm{t}\}_{1:K} = \{1, 2, ..., 10\}$, and spends the time interval $(\mathrm{t}_k, \mathrm{t}_{k+1})$ to train the policy, which consumes one second ($K = 10, N = 1$). The high frequency of interaction ($K = 10$) provides adequate data for precise future box position learning ($\mathrm{t} = 11$), yet a single policy update $(N = 1)$ may not approximate the optimal policy. Now, if the agent interacts with the environment four times, i.e. $\{\mathrm{t}\}_{1:K} = \{1, 4, 7, 10\}$ (see Figure 1(b)), it becomes feasible to train the policy over a duration of three seconds ($K = 4, N = 3$). A longer period of policy training $(N = 3)$ helps the agent in obtaining a near-optimal policy. However, limited observation data $(K = 4)$ and large time intervals ($\mathrm{t} \in \{11, 12, 13\}$) may lead to less accurate box predictions. This example underscores the practical importance of aligning the interaction time of the agent with the environment in non-stationary RL. Determining the optimal sequence $\{\mathrm{t}\}_{1:K}$ 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, the 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 $\{\mathtt{t}\}_{1:K}$, the existing formulations and methods for non-stationarity RL are insufficient. None of the existing works has adequately addressed this crucial aspect.

Our framework, ProST, provides a solution to finding the optimal $\{\mathtt{t}\}_{1:K}$ by computing a minimum solution to an upper bound on its performance measure. It proactively optimizes the time sequence by leveraging the agent tempo and the environment tempo. The ProST framework is divided into two components: future policy optimizer ($\texttt{OPT}_\pi$) and time optimizer ($\texttt{OPT}_\mathtt{t}$), 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 a *synchronizing tempo* framework as it finds a suboptimal training time by adjusting how many times the agent needs to update the policy (agent tempo) relative to how fast the environment changes (environment tempo). Our framework is general in the sense that it can be equipped with any common algorithm for policy update. Compared to the existing works [7–9], our approach achieves higher rewards and a more stable performance over time (see Figure 1(c) and Section 5).

We analyze the statistical and computational properties of ProST in a tabular MDP, which is named ProST-T. Our framework learns in a novel MDP, namely elapsed time-varying MDP, and quantifies its non-stationarity with a novel metric, namely time-elapsing variation budget, where both consider wall-clock time taken. We analyze 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 dynamic regret of $\texttt{OPT}_\mathtt{t}$ that computes a near-optimal policy in that model (Theorem 2, Proposition 2). We show that both regrets satisfy a sublinear rate with respect to the total number of episodes regardless of the agent tempo. More importantly, we obtain suboptimal training time by minimizing an objective that strikes a balance between the upper bounds of those two dynamic regrets, which reflect the tempos of the agent and the environment (Theorem 3). We find an interesting property that the future MDP model error of $\texttt{OPT}_\pi$ serves as a common factor on both regrets and show that the upper bound on the dynamic regret of ProST-T can be improved by a joint optimization problem of learning both different weights on observed data and a model (Theorem 4, Remark 1).

Finally, we introduce ProST-G, which is 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.

Notation

The sets of natural, real, and non-negative real numbers are denoted by $\mathbb{N}, \mathbb{R}$, and $\mathbb{R}_+$, respectively. 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 a variable X, we denote $\widehat{X}$ as a *forecasted* (or *predicted*) variable at the current time, and $\widetilde{X}$ as the observed value in the past. Also, for any time variable $\mathtt{t} > 0$ and $k \in \mathbb{N}$, we denote the time sequence $\{\mathtt{t}_1, \mathtt{t}_2, .., \mathtt{t}_k\}$ as $\{\mathtt{t}\}_{1:k}$, and variable X at time $\mathtt{t}_k$ as $X_{\mathtt{t}_k}$. We use the shorthand notation $X_{(k)}$(or $X^{(k)}$) for $X_{\mathtt{t}_k}$(or $X^{\mathtt{t}_k}$). We use the notation $\{x\}_{a:b}$ to denote a sequence of variables $\{x_a, x_{a+1}, ..., x_b\}$, and $\{x\}_{(a:b)}$ to represent a sequence of variables $\{x_{\mathtt{t}_a}, x_{\mathtt{t}_{a+1}}, ..., x_{\mathtt{t}_b}\}$. Given two variables x and y, let $x \vee y$ denote $\max(x, y)$, and $x \wedge y$ denote $\min(x, y)$. Given two complex numbers z_1 and z_2, we write $z_2 = W(z_1)$ if $z_2 e^{z_2} = z_1$, where W is the Lambert function. Given a variable x, the notation $a = \mathcal{O}(b(x))$ means that $a \leq C \cdot b(x)$ for some constant $C > 0$ that is independent of x, and the notation $a = \Omega(b(x))$ means that $a \geq C \cdot b(x)$ for some constant $C > 0$ that is independent of x. We have described the specific details in Appendix C.1.

2 Problem statement: Desynchronizing timelines

2.1 Time-elapsing Markov Decision Process

In this paper, we study a non-stationary Markov Decision Process (MDP) for which the transition probability and the reward change over time. 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). We introduce the new concept of **time-elapsing MDP**. It starts from the wall-clock time $\mathfrak{t} = 0$ to $\mathfrak{t} = T$, where T is fixed. The time-elapsing MDP at time $\mathfrak{t} \in [0, T]$ is defined as $\mathcal{M}_\mathfrak{t} \coloneqq \langle \mathcal{S}, \mathcal{A}, H, P_\mathfrak{t}, R_\mathfrak{t}, \gamma \rangle$, 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 the number of 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 the episode k. Similar to the existing non-stationary RL framework, the agent's objective is to learn 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 episode interactions, namely $\{\mathcal{M}_{\mathfrak{t}_1}, \mathcal{M}_{\mathfrak{t}_1}, ..., \mathcal{M}_{\mathfrak{t}_K}\}$, where the agent dedicates the time interval $(\mathfrak{t}_k, \mathfrak{t}_{k+1})$ for policy training and then obtains a sequence of suboptimal policies $\{\pi^{\mathfrak{t}_1}, \pi^{\mathfrak{t}_2}, ..., \pi^{\mathfrak{t}_K}\}$ to maximize a non-stationary policy evaluation metric, *dynamic regret*.

Dealing with time-elapsing MDP instead of conventional MDP raises an important question that should be addressed: within the time duration $[0, T]$, what time sequence $\{\mathfrak{t}\}_{1:K}$ yields favorable trajectory samples to obtain an optimal policy? This question is also related to the following: what is optimal value of K, i.e. the total number of episode that encompasses a satisfactory balance between the amount of observed trajectories and the accuracy of policy training? These interwined questions are concerned with an important aspect of RL, which is the computation of the optimal policy for a given $\mathfrak{t}_k$. In Section 4, we propose the `ProST` framework that computes a suboptimal K^* and its corresponding suboptimal time sequence $\{\mathfrak{t}^*\}_{1:K^*}$ based on the information of the environment's rate of change. In Section 3, we compute a near-optimal policy for $\{\mathfrak{t}^*\}_{1:K^*}$. Before proceeding with the above results, we introduce a new metric quantifying the environment's pace of change, referred to as time-elapsing variation budget.

2.2 Time-elapsing variation budget

Variation budget [10–12] is a metric to quantify the speed with which the environment changes. Driven by our motivations, we introduce a new metric imbued with real-time considerations, named *time-elapsing variation budget* $B(\Delta\mathfrak{t})$. Unlike the existing variation budget, which quantifies the environment's non-stationarity across episodes $\{1, 2, .., K\}$, our definition accesses it across $\{\mathfrak{t}_1, \mathfrak{t}_2, ..., \mathfrak{t}_K\}$, where the 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 another for reward function.

Definition 1 (Time-elapsing variation budgets)**.** *For a given sequence $\{\mathfrak{t}_1, \mathfrak{t}_2, .., \mathfrak{t}_K\}$, assume that the interval $\Delta\mathfrak{t}$ is equal to the policy training time Δ_π. We define two time-elapsing variation budgets $B_p(\Delta_\pi)$ and $B_r(\Delta_\pi)$ 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 using the time-elapsing variation budgets, we make the following assumption.

Assumption 1 (Drifting constants)**.** *There exist constants $c > 1$ and $\alpha_r, \alpha_p \geq 0$ such that $B_p(c\Delta_\pi) \leq c^{\alpha_p} B_p(\Delta_\pi)$ and $B_r(c\Delta_\pi) \leq c^{\alpha_r} B_r(\Delta_\pi)$. We call α_p and α_r the drifting constants.*

2.3 Suboptimal training time

Aside from the formal MDP framework, the agent can be informed of varying time-elapsing variation budgets based on the training time $\Delta_\pi \in (0, T)$ even within the same time-elapsing MDP. Intuitively, a short time Δ_π is inadequate to obtain a near-optimal policy, yet it facilitates frequent interactions with the environment, leading to a reduction in empirical model error due to a larger volume of data. On the contrary, a long time Δ_π may ensure obtaining a near-optimal policy but also introduces greater uncertainty in learning the environment. This motivates us to find a **suboptimal 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 a suboptimal $K^* = \lfloor T/\Delta_\pi^* \rfloor$, and a suboptimal time sequence where $\mathfrak{t}_k^* = \mathfrak{t}_1 + \Delta_\pi^* \cdot (k-1)$ for all $k \in [K^*]$. Our `ProST` framework computes the parameter Δ_π^*,

then sets $\{\mathfrak{t}^*\}_{1:K^*}$, and finally finds a *future* near-optimal policy for time $\mathfrak{t}^*_{k+1}$ at time $\mathfrak{t}^*_k$. In the next section, we first study how to approximate the one-episode-ahead suboptimal policy $\pi^{*,\mathfrak{t}_{k+1}}$ at time $\mathfrak{t}_k$ when $\{\mathfrak{t}\}_{1:K}$ is given.

3 Future policy optimizer

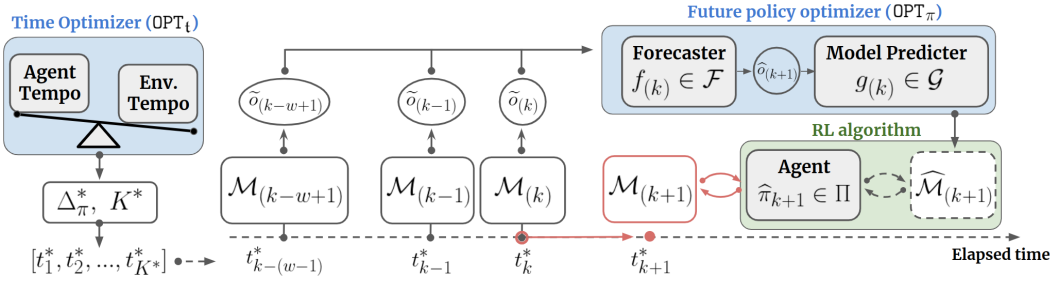

Figure 2: ProST framework

For given $\mathfrak{t}_k$ and $\mathfrak{t}_{k+1}$, the future policy optimizer (OPT$_\pi$), as a module of the ProST framework (Figure 2), computes a near-optimal policy for the future time $\mathfrak{t}_{k+1}$ at time $\mathfrak{t}_k$ via two procedures: (i) it first forecasts the future MDP model of time $\mathfrak{t}_{k+1}$ at time $\mathfrak{t}_k$ utilizing the MDP forecaster function, (ii) it then utilizes an arbitrary policy optimization algorithm within the forecasted MDP model OPT$_\pi$ to obtain a future near-optimal policy $\pi^{*,\mathfrak{t}_{k+1}}$.

3.1 MDP forecaster

Our ProST framework is applicable in an environment that meets the following assumption.

Assumption 2 (Observable non-stationary set $\mathcal{O}$)**.** *Assume that the non-stationarity of $\mathcal{M}_{\mathfrak{t}_k}$ is fully characterized by a non-stationary parameter $o_{\mathfrak{t}_k} \in \mathcal{O}$. Assume also that the agent observes a noisy non-stationary parameter $\tilde{o}_{\mathfrak{t}_k}$ at the end of episode $k \in [K]$ (at time $\mathfrak{t}_k$).*

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 those works for the estimation of $o_{(k)}$. Based on Assumption 2, we define the MDP forecaster function $g \circ f$ below.

Definition 2 (MDP forecaster $g \circ f$)**.** *Consider two function classes $\mathcal{F}$ and $\mathcal{G}$ such that $\mathcal{F} : \mathcal{O}^w \to \mathcal{O}$ and $\mathcal{G} : \mathcal{S} \times \mathcal{A} \times \mathcal{O} \to \mathbb{R} \times \Delta(\mathcal{S})$, where $w \in \mathbb{N}$. 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 $f_{(k)}$, acting as a non-stationarity forecaster, predicts a non-stationary parameter $\hat{o}_{(k+1)}$ at time t_{k+1} based on the last w observations given by the 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 number of used historical observations, denoted as w, by leveraging information from the environment (Section 4). Then, the function $g_{(k)}$, acting as a model predictor, predicts a reward $\widehat{R}_{(k+1)}(s,a)$ and a transition probability $\widehat{P}_{(k+1)}(\cdot|s,a)$ for time t_{k+1}, i.e., $(\widehat{R}_{(k+1)}, \widehat{P}_{(k+1)}) = g_{(k)}(s,a,\hat{o}_{k+1})$. Finally, the OPT$_\pi$ generates the estimated future MDP $\widehat{\mathcal{M}}_{(k+1)} = \langle \mathcal{S}, \mathcal{A}, H, \widehat{P}_{(k+1)}, \widehat{R}_{(k+1)}, \gamma \rangle$ associated with time t_{k+1}.

3.2 Finding future optimal policy

Now, consider an arbitrary RL algorithm provided by the user to obtain an optimal policy from the model $\widehat{\mathcal{M}}_{(k+1)}$. For a given time sequence $\{\mathfrak{t}\}_{1:K}$, the OPT$_\pi$ finds a near-optimal future policy as follows: (1) observe and forecast, (2) optimize using the future MDP model.

(1) Observe and forecast. At time $\mathfrak{t}_k$, the agent executes an episode k in the environment $\mathcal{M}_{(k)}$, completes its trajectory $\tau_{(k)}$, and observes the noisy non-stationary parameter $\hat{o}_{(k)}$ (Assumption 2). The algorithm then updates the function $f_{(k)}$ based on the last w observed parameters, and the

function $g_{(k)}$ with input from all previous trajectories. Following these updates, the MDP forecaster at time t_k predicts $\widehat{P}_{(k+1)}$ and $\widehat{R}_{(k+1)}$, thus creating the MDP model $\widehat{\mathcal{M}}_{(k+1)}$ for time $\mathfrak{t}_{k+1}$.

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

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

4 Time optimizer

4.1 Theoretical analysis

We now present our main theoretical contribution, which is regarding the time optimizer ($\texttt{OPT}_\mathfrak{t}$): computing a suboptimal policy training time Δ_π^* (the agent tempo). Our theoretical analysis starts with specifying the components of the $\texttt{OPT}_\pi$ optimizer, which we refer to as `ProST-T` (note that `-T` stands for an instance in the tabular setting). We employ the Natural Policy Gradient (NPG) with entropy regularization [17] as a policy update algorithm in OPT_π. We denote the entropy regularization coefficient as τ, the learning rate as η, the policy evaluation approximation gap arising due to finite samples as δ, and the past reference length for forecaster f as w. Without loss of generality, we assume that each policy iteration takes one second. The theoretical analysis is conducted within a tabular environment, allowing us to relax Assumption 2, which means that one can estimate non-stationary parameters by counting visitation of state and action pairs at time $\mathfrak{t}_k$, denoted as $n_{(k)}(s,a)$, rather than observing them. Additionally, we incorporate the exploration bonus term at time $\mathfrak{t}_k$ into $\widehat{R}_{(k+1)}$, denoted as $\Gamma_w^{(k)}(s,a)$, which is proportional to $\sum_{\tau=k-w+1}^{k}(n_{(\tau)}(s,a))^{-1/2}$ and aims to promote the exploration of states and actions that are visited infrequently.

We compute Δ_π^* by minimizing an upper bound on the `ProST-T`'s dynamic regret. The dynamic regret of `ProST-T` is characterized by the *model prediction error* that 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 $\mathfrak{t}_k$, the MDP forecaster predicts a model $\widehat{\mathcal{M}}_{(k+1)}$ and then we obtain a near-optimal policy $\widehat{\pi}^{(k+1)}$ based on $\widehat{\mathcal{M}}_{(k+1)}$. For each pair (s,a), we denote the state value function and the 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 $\widehat{Q}_h^{(k+1)}(s,a)$, respectively. We also denote the model prediction error associated with time $\mathfrak{t}_{k+1}$ calculated at time $\mathfrak{t}_k$ as $\iota_h^{(k+1)}(s,a)$, which is defined as*

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

We now derive an upper bound on the `ProST-T` dynamic regret. We expect the upper bound to be likely controlled by two factors: the error of the MDP forecaster's prediction of the future MDP model and the error of the NPG algorithm due to approximating the optimal policy within an estimated future MDP model. This insight is clearly articulated in the next theorem.

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 $\bar{\iota}_\infty^K := \sum_{k=1}^{K-1}\|\bar{\iota}_\infty^{k+1}\|_\infty$, where ι_H^K is a data-dependent error. For a 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 at least $1-p/2$ as follows:*

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

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

Specifically, the upper bound is composed of two terms: $\mathfrak{R}_I$ that originates from the MDP forecaster error between $\mathcal{M}_{(k+1)}$ and $\widehat{\mathcal{M}}_{(k+1)}$, and $\mathfrak{R}_{II}$ that arises due to the suboptimality gap between $\pi^{*,(k+1)}$ and $\widehat{\pi}^{(k+1)}$. Theorem 1 clearly demonstrates that a prudent construction of the MDP

forecaster that controls the model prediction errors and the selection of the agent tempo Δ_π is significant in guaranteeing sublinear rates for $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$. To understand the role of the environment tempo in $\mathfrak{R}_I$, we observe that the MDP forecaster utilizes w previous observations, which inherently encapsulates the environment tempo. We expect the model prediction errors, at least in part, to be controlled by the environment tempo $B(\Delta_\pi)$, so that a trade-off between two tempos can be framed as the trade-off between $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$. Hence, it is desirable to somehow minimize the upper bound with respect to Δ_π to obtain a solution, denoted as Δ_π^* , which strikes a balance between $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$.

4.1.1 Analysis of $\mathfrak{R}_{II}$

A direct analysis of the upper bound $\mathfrak{R}_I + \mathfrak{R}_{II}$ is difficult since its dependence on K is not explicit. To address this issue, we recall that an optimal Δ_π^* should be a natural number that guarantees the sublinearity of both $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$ with respect to the total number of episodes K. We first compute a set $\mathbb{N}_{II} \subset \mathbb{N}$ that includes those values of Δ_π that guarantee the sublinearity of $\mathfrak{R}_{II}$, and then compute a set $\mathbb{N}_I \subset \mathbb{N}$ that guarantees the sublinearity of $\mathfrak{R}_I$. Finally, we solve for Δ_π^* in the common set $\mathbb{N}_I \cap \mathbb{N}_{II}$.

Proposition 1 (Δ_π bounds for sublinear $\mathfrak{R}_{II}$)**.** *A total step H is given by MDP. For a number $\epsilon > 0$ such that $H = \Omega\left(\log\left((\widehat{r}_{max} \vee r_{max})/\epsilon\right)\right)$, we choose δ, τ, η to satisfy $\delta = \mathcal{O}(\epsilon)$, $\tau = \Omega\left(\epsilon/\log|\mathcal{A}|\right)$ and $\eta \leq (1-\gamma)/\tau$, where $\widehat{r}_{max}$ and r_{max} are the maximum reward of the forecasted model and the maximum reward of the environment, respectively. Define $\mathbb{N}_{II} := \{n \mid n > \frac{1}{\eta\tau}\log\left(\frac{C_1(\gamma+2)}{\epsilon}\right), n \in \mathbb{N}\}$, where C_1 is a constant. Then $\mathfrak{R}_{II} \leq 4\epsilon(K-1)$ for all $\Delta_\pi \in \mathbb{N}_{II}$.*

As a by-product of Proposition 1, the sublinearity of $\mathfrak{R}_{II}$ can be realized by choosing $\epsilon = \mathcal{O}((K-1)^{\alpha-1})$ for any $\alpha \in [0,1)$, which suggests that a tighter upper bound on $\mathfrak{R}_{II}$ requires a smaller ϵ and subsequently 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 Analysis of $\mathfrak{R}_I$

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

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

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

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 a brief sketch of how `SW-LSE` makes the environment tempo appear in the upper bound, we outline that the model prediction errors are upper-bounded by two forecaster errors, namely $P_{(k+1)}$ - $\widehat{P}_{(k+1)}$ and $R_{(k+1)} - \widehat{R}_{(k+1)}$, along with the visitation count $n_{(k)}(s,a)$. Then, the `SW-LSE` algorithm provides a solution $\left(\widehat{P}_{(k+1)}, \widehat{R}_{(k+1)}\right)$ as a closed form of linear combinations of past w estimated values $\{\widetilde{P}, \widetilde{R}\}_{(k-w+1:w)}$. Finally, employing the Cauchy inequality and triangle inequality, we derive two forecasting errors that are upper-bounded by the environment tempo. For the final step before obtaining a suboptimal Δ_π^*, we compute $\mathbb{N}_I$ that guarantees the sublinearity of $\mathfrak{R}_I$.

Proposition 2 (Δ_π bounds for sublinear $\mathfrak{R}_I$)**.** *Denote $B(1)$ as the environment tempo when $\Delta_\pi = 1$, which is a summation over all time steps. Assume that the environment satisfies $B_r(1) + B_p(1)\hat{r}_{max}/(1-\gamma) = o(K)$ and we choose $w = \mathcal{O}((K-1)^{2/3}/(C_I[B(\Delta_\pi)])^{2/3})$. Define the set $\mathbb{N}_I$ to be $\{n \mid n < K,\ n \in \mathbb{N}\}$. Then $\mathfrak{R}_I$ is upper-bounded as $\mathfrak{R}_I = \mathcal{O}\left(C_I[B(\Delta_\pi)]^{1/3}(K-1)^{2/3}\sqrt{\log((K-1)/C_I[B(\Delta_\pi)])}\right)$ and also satisfies a sublinear upper bound, provided that $\Delta_\pi \in \mathbb{N}_I$.*

The upper bound on the environment tempo $B(1)$ in proposition 2 is aligned with our expectation that dedicating an excessively long time to a single iteration may not allow for an effective policy approximation, thereby hindering the achievement of a sublinear dynamic regret. Furthermore, 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 Suboptimal tempo Δ_π^*

So far, we have shown that an upper bound on the ProST dynamic regret is composed of two terms $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$, which are characterized by the environment tempo and the agent tempo, respectively. Now, we claim that a suboptimal tempo that minimizes ProST's dynamic regret could be obtained by the optimal solution $\Delta_\pi^* = \arg\min_{\Delta_\pi \in \mathbb{N}_I \cap \mathbb{N}_{II}} (\mathfrak{R}_I^{\max} + \mathfrak{R}_{II}^{\max})$, where $\mathfrak{R}_I^{\max}$ and $\mathfrak{R}_{II}^{\max}$ denote the upper bounds on $\mathfrak{R}_I$ and $\mathfrak{R}_{II}$. We show that Δ_π^* strikes a balance between the environment tempo and the agent tempo since $\mathfrak{R}_I^{\max}$ is a non-decreasing function of Δ_π and $\mathfrak{R}_{II}^{\max}$ is a non-increasing function of Δ_π. Theorem 3 shows that the optimal tempo Δ_π^* depends on the environment's drifting constants introduced in Assumption 1.

Theorem 3 (Suboptimal tempo Δ_π^*)**.** *Let* $k_{\texttt{Env}} = (\alpha_r \vee \alpha_p)^2 C_I[B(1)]$, $k_{\texttt{Agent}} = \log(1/(1-\eta\tau))C_1(K-1)(\gamma+2)$. *Consider three cases:* ***case1****:* $\alpha_r \vee \alpha_p = 0$, ***case2****:* $\alpha_r \vee \alpha_p = 1$, ***case3****:* $0 < \alpha_r \vee \alpha_p < 1$ *or* $\alpha_r \vee \alpha_p > 1$. *Then* Δ_π^* *depends on the environment's drifting constants as follows:*

- *Case1:* $\Delta_\pi^* = T$.
- *Case2:* $\Delta_\pi^* = \log_{1-\eta\gamma}(k_{\texttt{Env}}/k_{\texttt{Agent}}) + 1$.
- *Case3:* $\Delta_\pi^* = \exp\left(-W\left[-\frac{\log(1-\eta\tau)}{\max(\alpha_r,\alpha_p)-1}\right]\right)$, *provided that the parameters are chosen so that* $k_{\texttt{Agent}} = (1-\eta\tau)k_{\texttt{Env}}$.

4.2 Improving MDP forecaster

Determining a suboptimal tempo by minimizing an upper bound on $\mathfrak{R}_I + \mathfrak{R}_{II}$ can be improved by using a tighter upper bound. In Proposition 1, we focused on the Q approximation gap δ to provide a justifiable upper bound on $\mathfrak{R}_I + \mathfrak{R}_{II}$. It is important to note that the factor δ arises not only from the finite sample trajectories as discussed in [21], but also from the forecasting error between $\mathcal{M}_{(k+1)}$ and $\widehat{\mathcal{M}}_{(k+1)}$. It is clear that the MDP forecaster establishes a lower bound on δ denoted as $\delta_{\min}$, which in turn sets a lower bound 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 a further investigation to improve the accuracy of the forecaster is necessary for a better bounding on $\mathfrak{R}_I + \mathfrak{R}_{II}$.

Our approach to devising a precise MDP forecaster is that, instead of *selecting* the past reference length w as indicated in Proposition 2, we set $w = k$, implying the utilization of all past observations. However, we address this by solving an additional optimization problem, resulting in a tighter bound on $\mathfrak{R}_I$. We propose a method that adaptively assigns different weights $q \in \mathbb{R}_+^k$ to the previously observed non-stationary parameters up to time t_k, which reduces the burden of choosing w. Hence, we further analyze $\mathfrak{R}_I$ through the utilization of the Weighted regularized Least-Squares Estimator (W-LSE) [22]. Unlike SW-LSE, W-LSE does not necessitate a predefined selection of w, but it instead engages in a joint optimization procedure involving the data weights q and the future model $\left(\widehat{P}_{(k+1)}, \widehat{R}_{(k+1)}\right)$. To this end, we define the forecasting reward model error as $\Delta_k^r(s,a) = \left|\left(R_{(k+1)} - \widehat{R}_{(k+1)}\right)(s,a)\right|$ and the 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 with f=W-LSE)**.** *By setting the exploration bonus* $\Gamma_{(k)}(s,a) = \frac{1}{2}\Delta_k^r(s,a) + \frac{\gamma\tilde{r}_{max}}{2(1-\gamma)}\Delta_k^p(s,a)$, *it holds that*

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

Remark 1 (Tighter $\mathfrak{R}_{\mathcal{I}}$ upper bound with f = W-LSE)**.** *If the optimization problem of W-LSE is feasible, then the optimal data weight $q^\star$ provides tighter bounds for Δ_k^r and Δ_k^p in comparison to SW-LSE, consequently leading to a tighter upper bound for $\mathfrak{R}_{\mathcal{I}}$. We prove in Lemmas 4 and 6 in Appendix D.3 that $\bar{\iota}_\infty^K$ and $-\iota_H^K$ are upper-bounded in terms of Δ_k^r and Δ_k^p.*

4.3 ProST-G

The theoretical analysis outlined above serves as a 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, named ProST-G, which particularly extends the investigation in Section 4.2. ProST-G optimizes a policy with the soft actor-critic (SAC) algorithm [23], utilizes the 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 evaluate 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 make the rewards in the three environments non-stationary by altering the agent's desired directions. The 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. The non-stationary parameter o_k is generated from the sine function with five different speeds and from the real data A and B. We then measure the 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 outcomes of the experimental results are 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 Appendix E.3. In most cases, ProST-G outperforms MBPO in terms of rewards, highlighting the adaptability of the 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 a 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 Theorem 3.

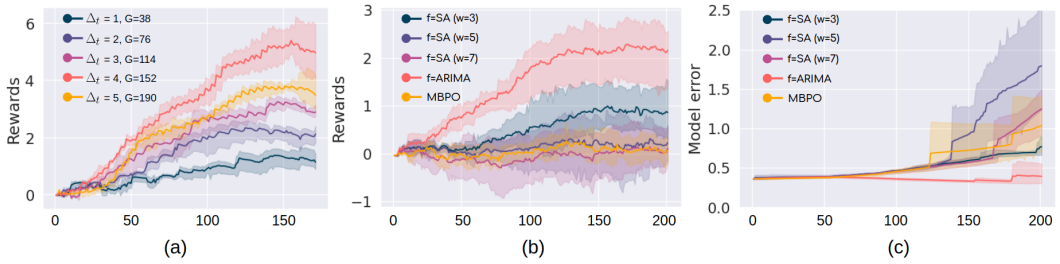

Figure 3: (a) Optimal Δ_π^*; (b) Different forecaster f (ARIMA, SA); (c) The Mean squared Error (MSE) model loss of `ProST-G` with four different forecasters (ARIMA and three SA) and the MBPO. The x-axis in each figure shows the episodes.

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

Functions f, g. We investigate the effect of the forecaster f's accuracy on the framework using two distinct functions: ARIMA and a simple average (SA) model, each tested with three different the values of w. Figure 3(b) shows the average rewards of the SA model with $w \in \{3, 5, 7\}$ and ARIMA model (four solid lines). The shaded area is 95 % the confidence area among 4 different speeds $\{1, 2, 3, 4\}$. Figure 3(c) shows the corresponding model error. Also, we investigate the effect of the different model predictor g by comparing MBPO (reactive-model) and `ProST-G` with f =ARIMA (proactive-model) in Figure 3(c). The high returns from `ProST-G` with f = ARIMA, compared to those from MBPO, empirically support that the forecasting component of the **ProST** framework can provide a satisfactory adaptability to the baseline algorithm that is equipped with. Also, Figures 3(b) and 3(c) provide empirical evidence that the accuracy of f is contingent on the sliding window size, thereby impacting the model accuracy and subsequently influencing the agent's performance.

7 Conclusion

This work offers the first study on the important issue of time synchronization for non-stationary RL. To this end, we introduce the concept of the tempo of adaptation in a non-stationary RL, and obtain a suboptimal training time. We propose a Proactively Synchronizing Tempo (ProST) framework, together with two specific instances `ProST-T` and `ProST-G`. The proposed method adjusts an agent's tempo to match the tempo of the environment to handle non-stationarity through both theoretical analysis and empirical evidence. The ProST framework provides a new avenue to implement reinforcement learning in the real world by incorporating the concept of adaptation tempo.

As a future work, it is important to generalize the proposed framework to learn a safe guarantee policy in a non-stationary RL by considering the adaptation tempo of constraint violations [25, 26]. Another generalization is to introduce an alternative dynamic regret metric, enabling a fair performance comparison among agents, even when they have varying numbers of total episodes. Another future work is to find an optimal 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 [27, 28].

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. In *International Conference on Machine Learning*, 2019.

[4] Luisa Zintgraf, Sebastian Schulze, Cong Lu, Leo Feng, Maximilian Igl, Kyriacos Shiarlis, Yarin Gal, Katja Hofmann, and Shimon Whiteson. Varibad: Variational bayes-adaptive deep rl via meta-learning. *Journal of Machine Learning Research*, 22(289):1–39, 2021.

[5] Zhuangdi Zhu, Kaixiang Lin, Anil K. Jain, and Jiayu Zhou. Transfer learning in deep reinforcement learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45 (11):13344–13362, 2023.

[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. In *International Conference on Learning Representations*, 2018.

[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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(6):7405–7413, 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, and Tie-Yan Liu. An adaptive deep rl method for non-stationary environments with piecewise stable context. In *Advances in Neural Information Processing Systems*, volume 35, pages 35449–35461, 2022.

[14] Biwei Huang, Fan Feng, Chaochao Lu, Sara Magliacane, and Kun Zhang. Adarl: What, where, and how to adapt in transfer reinforcement learning. In *International Conference on Learning Representations*, 2022.

[15] Fan Feng, Biwei Huang, Kun Zhang, and Sara Magliacane. Factored adaptation for non-stationary reinforcement learning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 31957–31971, 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] 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.

[27] 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. *International Conference on Learning Representations*, 2022.

[28] 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, 2021.

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

[30] 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.

[31] Yuhao Ding and Javad Lavaei. Provably efficient primal-dual reinforcement learning for cmdps with non-stationary objectives and constraints. AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. ISBN 978-1-57735-880-0.

[32] 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.

[33] 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. *International Conference on Learning Representations*, 2016.

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

[35] Yuping Luo, Huazhe Xu, Yuanzhi Li, Yuandong Tian, Trevor Darrell, and Tengyu Ma. Algorithmic framework for model-based deep reinforcement learning with theoretical guarantees. *International Conference on Learning Representations*, 2019.

[36] 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.

[37] 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.

[38] 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.

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

2. Experiment details

To motivate our proposed meta-framework via 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 $\eta \in \{0.001, 0.003, 0.005, 0.007\}$ and the entropy regularized parameter $\tau = 0.1$, where the shaded area of the three baselines is 95 % confidence area among 4 different learning rates. The `PTM-T` was trained with the model rollout length $\widehat{H} \in \{50, 60\}$, policy update iteration number $G \in \{10, 50\}$, entropy regularized parameter $\tau = 0.1$, Q-learning update parameter $\alpha_Q \in \{0.7, 0.9, 0.99\}$, and the learning rate $\eta = 0.001$. The shaded area of `PTM-T` is 95 % confidence area among the 12 different cases above. All four algorithms share the same agent's policy network structure.

B Related Works

Existing methods for non-stationary environments can be grouped into three categories: 1) shoehorning: directly using established frameworks for stationary MDPs by 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, 29], online model updates, or latent factor identification [4, 13–16]; and 3) anticipating future changes by forecasting policy gradients or value functions [7, 30, 20, 10, 31].

The advantage of the model-free method is its computational efficiency, allowing for direct learning of complex policies from raw data [32, 33], while the advantage of the model-based method is its data efficiency, allowing one to learn fast by learning how the environment works [34, 35]. 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 [36, 37] 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, 37] focused 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. [30] proposed adaptive Q-learning with a restart strategy and established its near-optimal dynamic regret bound. In addition, [36] proposed two model-free policy optimization algorithms based on the restart strategy and showed that dynamic regret satisfies polynomial space and time complexities. However, the provable model-free methods in [30, 36] still lack empirical evidence and adaptability in complex environments. Furthermore, since the agent can execute a policy in a fixed environment only once due to the non-stationarity of the environment, 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. The work [14] learned the model change factors and their representation in heterogeneous domains with varying reward functions and dynamics. However, it has restrictions for use in non-stationary environments, meaning that it 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] brought 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. The recent works [20, 10, 31] 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. First, we denote the state and action at the wall-clock time $\mathfrak{t}_k$ of step h as $s_h^{\mathfrak{t}_k}$ and $a_h^{\mathfrak{t}_k}$, respectively. As mentioned in the main paper, we interchangeably use the symbols $s_h^{(k)}$ and $a_h^{(k)}$ for $s_h^{\mathfrak{t}_k}$ and $a_h^{\mathfrak{t}_k}$. At the wall-clock time $\mathfrak{t}_k$, the agent starts from an initial state $s_0^{\mathfrak{t}_k} \sim \rho$. At step $h \in [H]$ of the episode k, the agent takes the action $a_h^{\mathfrak{t}_k} = \pi^{\mathfrak{t}_k}(\cdot|s_h^{\mathfrak{t}_k})$ from the current state $s_h^{\mathfrak{t}_k}$. The agent then receives the reward $r_h^{\mathfrak{t}_k} \sim R_{\mathfrak{t}_k}(s_h^{\mathfrak{t}_k}, a_h^{\mathfrak{t}_k})$ and moves to the next state $s_{h+1}^{\mathfrak{t}_k} \sim P_{\mathfrak{t}_k}(s_{h+1}^{\mathfrak{t}_k}|s_h^{\mathfrak{t}_k}, a_h^{\mathfrak{t}_k})$. The trajectory ends when the agent reaches $s_H^{\mathfrak{t}_k}$.

Future MDP $\widehat{\mathcal{M}}_{t_{k+1}}$. Our work creates a one-episode-ahead MDP $\widehat{\mathcal{M}}_{t_{k+1}}$ based on the observed data from the 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 $\widehat{P}_{\mathfrak{t}_{k+1}}$ and $\widehat{R}_{t_{k+1}}$ are the *forecasted* future transition probability and reward function, respectively. As mentioned in the main paper, 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 the state, action, and policy in $\widehat{\mathcal{M}}_{t_{k+1}}$ as $\widehat{s}_h^{t_{k+1}}, \widehat{a}_h^{t_{k+1}}, \widehat{\pi}^{t_{k+1}}$, or equivalently $\widehat{s}_h^{(k+1)}, \widehat{a}_h^{(k+1)}, \widehat{\pi}^{(k+1)}$, respectively. We elaborate our main methodology in Section 3.

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

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

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

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

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

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

As mentioned in the main paper, we simplify the symbols $V^{\pi,\mathfrak{t}_k}, Q^{\pi,\mathfrak{t}_k}, \widehat{V}^{\pi,\mathfrak{t}_{k+1}}, \widehat{Q}^{\pi,\mathfrak{t}_{k+1}}$ as $V^{\pi,(k)}, Q^{\pi,(k)}, \widehat{V}^{\pi,(k+1)}, \widehat{Q}^{\pi,(k+1)}$.

Dynamic regret. Aside from stationary MDPs, the agent aims to maximize the cumulative expected reward throughout the K episodes by adopting a sequence of policies $\{\pi^{\mathfrak{t}_k}\}_{1:K}$. In non-stationary MDPs, the optimality of the policies is evaluated in terms of the dynamic regret $\mathfrak{R}\big(\{\pi^{\mathfrak{t}_k}\}_{1:K}, K\big)$ defined as

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

where $V^{*,\mathfrak{t}_k} (= V^{\pi^{*,\mathfrak{t}_k},\mathfrak{t}_k})$ denotes the optimal state value function under the optimal policy $\pi^{*,\mathfrak{t}_k}$ at the wall-clock time $\mathfrak{t}_k$ (episode k) and $V^{\pi^{\mathfrak{t}_k},\mathfrak{t}_k}$ denotes the state value with agent's k^{th} episode's policy π^k. Dynamic regret is a stronger evaluation than the standard static regret that considers the optimality of a single policy over all episodes.

State value and state-action value functions at step h. We denote the state value function and the state-action value function for any policy π at *step* h of the wall-clock time $\mathfrak{t}_k$ as $V_h^{\pi,\mathfrak{t}_k}$ and $Q_h^{\pi,\mathfrak{t}_k}$, respectively. We define

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

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

Then, the corresponding Bellman equation is

$$Q_h^{\pi,\mathfrak{t}_k}(s,a) = \big(R_{\mathfrak{t}_k} + \gamma P_{\mathfrak{t}_k} V_{h+1}^{\pi,\mathfrak{t}_k}\big)(s,a), \;\; V_h^{\pi,\mathfrak{t}_k}(s) = \big\langle Q_h^{\pi,\mathfrak{t}_k}(s,\cdot), \pi(\cdot|s)\big\rangle_{\mathcal{A}}, \;\; V_H^{\mathfrak{t}_k,\pi}(s) = 0 \;\forall s \in \mathcal{S} \tag{C.2}$$

where $\big(P_{\mathfrak{t}_k} f\big)(s,a) \coloneqq \mathbb{E}_{s' \sim P^{\mathfrak{t}_k}(\cdot|s,a))}[f(s')]$ for every function $f : \mathcal{S} \to \mathbb{R}$.

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

$$Q_h^{*,\mathfrak{t}_k}(s,a) = \big(R_{\mathfrak{t}_k} + \gamma P_{\mathfrak{t}_k} V_{h+1}^{*,\mathfrak{t}_k}\big)(s,a), \;\; V_h^{*,\mathfrak{t}_k}(s) = \big\langle Q_h^{*,\mathfrak{t}_k}(s,\cdot), \pi^{*,\mathfrak{t}_k}(\cdot|s)\big\rangle_{\mathcal{A}}, \tag{C.3}$$

$$\pi^{*,\mathfrak{t}_k}(s) = \max_a Q_h^{*,\mathfrak{t}_k}(s,a).$$

We also denote the *forecasted* state value at the wall-clock time $\mathfrak{t}_{k+1}$ of step h when the agent is stated at time $\mathfrak{t}_k$ as $\widehat{V}_h^{\pi,\mathfrak{t}_{k+1}}$ and the *forecasted* state-action value as $\widehat{Q}_h^{\pi,\mathfrak{t}_{k+1}}$ in a forecasted MDP $\widehat{\mathcal{M}}_{\mathfrak{t}_{k+1}}$. We define

$$\widehat{V}_h^{\pi,\mathfrak{t}_{k+1}}(s) \coloneqq \mathbb{E}_{\widehat{\mathcal{M}}_{\mathfrak{t}_{k+1}},\pi}\Big[\sum_{i=h}^{H-1} \gamma^{i-h} \widehat{r}_i^{\mathfrak{t}_{k+1}} \mid \widehat{s}_h^{\mathfrak{t}_{k+1}} = s\Big], \tag{C.4}$$

$$\widehat{Q}_h^{\pi,\mathfrak{t}_{k+1}}(s,a) \coloneqq \mathbb{E}_{\widehat{\mathcal{M}}_{\mathfrak{t}_{k+1}},\pi}\Big[\sum_{i=h}^{H-1} \gamma^{i-h} \widehat{r}_i^{\mathfrak{t}_{k+1}} \mid \widehat{s}_h^{\mathfrak{t}_{k+1}} = s, \widehat{a}_h^{\mathfrak{t}_{k+1}} = a\Big]. \tag{C.5}$$

Then, the Bellman equation is given by

$$\widehat{Q}_h^{\pi,\mathfrak{t}_{k+1}}(s,a) = \big(\widehat{R}_{\mathfrak{t}_{k+1}} + \gamma \widehat{P}_{\mathfrak{t}_{k+1}} \widehat{V}_{h+1}^{\pi,\mathfrak{t}_{k+1}}\big)(s,a), \;\; \widehat{V}_h^{\pi,\mathfrak{t}_{k+1}}(s) = \big\langle \widehat{Q}_h^{\pi,\mathfrak{t}_{k+1}}(s,\cdot), \pi(\cdot|s)\big\rangle_{\mathcal{A}},$$

$$\widehat{V}_H^{\pi,\mathfrak{t}_{k+1}}(s) = 0 \;\forall s \in \mathcal{S}. \tag{C.6}$$

We denote the *future* optimal policy of the *future* value function $\widehat{V}^{\pi,\mathfrak{t}_{k+1}}$ as $\widehat{\pi}^{*,\mathfrak{t}_{k+1}}$. Then the Bellman equation also holds for $\widehat{Q}_h^{\pi,\mathfrak{t}_{k+1}}(s)$ and $\widehat{V}_h^{\pi,\mathfrak{t}_{k+1}}(s)$ as follows:

$$\widehat{Q}_h^{*,\mathfrak{t}_{k+1}}(s,a) = \big(\widehat{R}_{\mathfrak{t}_{k+1}} + \gamma \widehat{P}_{\mathfrak{t}_{k+1}} \widehat{V}_{h+1}^{*,\mathfrak{t}_{k+1}}\big)(s,a), \;\; \widehat{V}_h^{*,\mathfrak{t}_{k+1}}(s) = \big\langle \widehat{Q}_h^{*,\mathfrak{t}_{k+1}}(s,\cdot), \widehat{\pi}^{*,\mathfrak{t}_{k+1}}(\cdot|s)\big\rangle_{\mathcal{A}},$$

$$\widehat{\pi}^{*,\mathfrak{t}_{k+1}}(s) = \max_a \widehat{Q}_h^{*,\mathfrak{t}_{k+1}}(s,a). \tag{C.7}$$

As mentioned in the main paper, we simplify the notations $V_h^{\pi,\mathfrak{t}_k}, Q_h^{\pi,\mathfrak{t}_k}, \widehat{V}_h^{\pi,\mathfrak{t}_{k+1}}, \widehat{Q}_h^{\pi,\mathfrak{t}_{k+1}}$ as $V_h^{\pi,(k)}, Q_h^{\pi,(k)}, \widehat{V}_h^{\pi,(k+1)}, \widehat{Q}_h^{\pi,(k+1)}$.

Unnormalized (discounted) occupancy measure. We define the unnormalized (discounted) occupancy measure $\nu_{s_0,a_0}^{\pi,\mathfrak{t}_k} \in \Delta_{1/(1-\gamma)}(\mathcal{S}\times\mathcal{A})$ at wall-clock time $\mathfrak{t}_k$ (episode k) for a given policy π together with an initial state s_0 and the action a_0 as

$$\nu_{s_0,a_0}^{\pi,\mathfrak{t}_k}(s,a) \coloneqq \sum_{h=0}^{\infty} \gamma^h \mathbb{P}(s_h = s, a_h = a \mid s_0, a_0 \,;\, \pi, P_{\mathfrak{t}_k}) \;\;,\;\; \forall (s,a) \in \mathcal{S}\times\mathcal{A} \tag{C.8}$$

where $\mathbb{P}(s_h = s, a_h = a \mid s_0, a_0 \,;\, \pi, P^{\mathfrak{t}_k})$ is the probability of visiting (s,a) at step h when following policy π from (s_0,a_0) with the transition probability $P_{\mathfrak{t}_{k+1}}$.

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

$$\widehat{\nu}_{s_0,a_0}^{\pi,\mathfrak{t}_{k+1}}(s,a) \coloneqq \sum_{h=0}^{\infty} \gamma^h \mathbb{P}(s_h = s, a_h = a \mid s_0, a_0, \pi, \widehat{P}_{\mathfrak{t}_{k+1}}) \;\;,\; \forall (s,a) \in \mathcal{S}\times\mathcal{A} \tag{C.9}$$

where the probability is defined in a forecasted environment with $\widehat{P}_{\mathfrak{t}_{k+1}}$.

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

$$\bar{\iota}_\infty^{\mathfrak{t}_{k+1}}(s,a) \coloneqq \left(R_{\mathfrak{t}_{k+1}} + \gamma P_{\mathfrak{t}_{k+1}} \widehat{V}_\infty^{*,\mathfrak{t}_{k+1}} - \widehat{Q}_\infty^{*,\mathfrak{t}_{k+1}}\right)(s,a), \tag{C.10}$$

$$\iota_h^{\mathfrak{t}_{k+1}}(s,a) \coloneqq \left(R_{\mathfrak{t}_{k+1}} + \gamma P_{\mathfrak{t}_{k+1}} \widehat{V}_{h+1}^{\widehat{\pi}^{\mathfrak{t}_{k+1}},\mathfrak{t}_{k+1}} - \widehat{Q}_h^{\widehat{\pi}^{\mathfrak{t}_{k+1}},\mathfrak{t}_{k+1}}\right)(s,a). \tag{C.11}$$

As mentioned in the main paper, we allow $\bar{\iota}_\infty^{\mathfrak{t}_{k+1}}(s,a)$ and $\iota_h^{\mathfrak{t}_{k+1}}(s,a)$ to be interchangeably expressed by the symbols $\bar{\iota}_\infty^{(k+1)}(s,a)$ and $\iota_h^{(k+1)}(s,a)$.

Local time-elapsing variation budget. Aside from the time-elapsing variation budget, we define the *local* time-elapsing variation budgets $B_p^{(k-w:k)}$ and $B_r^{(k-w:k)}$ that quantifie how fast the environment changes over wall-clock times $\{\mathfrak{t}_{k-w+1}, \mathfrak{t}_{k+1}, ..., \mathfrak{t}_k\}$ where $k-w, k \in [K]$:

$$B_p^{(k-w+1:k)}(\Delta_\pi) \coloneqq \sum_{\tau=k-w+1}^{k} \sup_{s,a} \|P_{\mathfrak{t}_{\tau+1}}(\cdot\,|s,a) - P_{\mathfrak{t}_\tau}(\cdot\,|s,a)\|_1,$$

$$B_r^{(k-w+1:k)}(\Delta_\pi) \coloneqq \sum_{\tau=k-w+1}^{k} \sup_{s,a} |R_{\mathfrak{t}_{k+1}}(s,a) - R_{\mathfrak{t}_k}(s,a)|.$$

D Proof of Theoretical Analysis

D.1 Preliminary for ProST-T and theoretical analysis

In this subsection, we elaborate on the `ProST-T`'s environment setting and its components f, g.

D.1.1 Environment setting

We consider the tabular environment have the following properties:

1. First, $P_{(k)}$ and $R_{(k)}$ are represented by the inner products of the feature functions $\psi : \mathcal{S}\times\mathcal{S}\times\mathcal{A} \to \mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|}, \varphi : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ and the non-stationary variables $o_{(k)}^p, o_{(k)}^r \in \mathcal{O}$, respectively, where $o_{(k)}^p : \mathcal{S}\times\mathcal{S}\times\mathcal{A} \to \mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|}$ and $o_{(k)}^r : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$. That is, $P_{(k)} =< \psi, o_{(k)}^p >$ and $R_{(k)} =< \varphi, o_{(k)}^r >$.

2. Second, the agent estimates $o^p_{(k)}$ and $o^r_{(k)}$ rather than observing them. More specifically, we consider the non-stationary variable set $\mathcal{O}$ to be the set $\{P_{(k)}\}_{1:K}, \{R_{(k)}\}_{1:K}$. The agent then attempts to *estimate* o_k (denote $P_{(k)}$ as $o^p_{(k)}$ and $R_{(k)}$ as $o^r_{(k)}$) through its w lastest trajectories, where Assumption 2 does not need to be satisfied 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 last w trajectories, i.e., $\tau_{k-(w-1):k}$.

We elaborate on the above two settings below:

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

Let us define a set of one-hot reward vectors over all states and the action space, namely $\mathbb{1}_r \coloneqq \{\varphi^y \in \{0,1\}^{|\mathcal{S}||\mathcal{A}|} \mid \sum_{i=1}^{|\mathcal{S}||\mathcal{A}|} \varphi^y_i = 1\}$, and similarly define a set of one-hot transition probability vectors, namely $\mathbb{1}_p \coloneqq \{\psi^y \in \{0,1\}^{|\mathcal{S}|^2|\mathcal{A}|} \mid \sum_{i=1}^{|\mathcal{S}|^2|\mathcal{A}|} \psi^y_i = 1\}$. We then define one-to-one functions φ and ψ such that $\varphi : \mathcal{S} \times \mathcal{A} \to \mathbb{1}_r$ and $\psi : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathbb{1}_p$. Namely, $\varphi(s,a)(\psi(s',s,a))$ is a one-hot vector such that the $(i)^{th}$ entry equals 1. We use 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 similarly $\psi^k_h = \psi(s^{(k)}_{h+1}, s^{(k)}_h, a^{(k)}_h)$.

Then, we set $\mathcal{O} = \{P_{(k)}, R_{(k)}\}_{k=1}^{\infty}$ in `ProST-T`. Also, we set o_k to consist of two parameters as $o_k = (o^p_{(k)}, o^r_{(k)})$. We define a function $o^p_{(k)} \coloneqq \{o : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathbb{R}^{|\mathcal{S}|^2|\mathcal{A}|} \mid o(s',s,a) = P_{(k)}(s'|s,a),\ \forall (s',s,a)\}$ and 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 $P_{(k)}$ and $R_{(k)}$ can be constructed by the inner products of the stationary functions φ and ψ and the unknown non-stationary parameters $o^p_{(k)}$ and $o^r_{(k)}$ as follows,

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

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

For notational simplicity, we use $< \psi, o^p_{(k)} >$ and $< \varphi, o^r_{(k)} >$ to show the inner products of the functions $\psi, o^p_{(k)}$ and $\varphi, o^r_{(k)}$, respectively. Therefore, $P_{(k)} =< \psi, o^p_{(k)} >$ and $R_{(k)} =< \varphi, o^r_{(k)} >$.

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

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

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

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

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

where $n_{(k)}(s,a)$ denotes visitation count of state s under action a through trajectory $\tau_{(k)}$ and $n_{(k)}(s,a,s')$ denotes visitation count of state s under action a and subsequent next state s' through trajectory. We denote $\widehat{o}^p_{k,h} = \widehat{o}^p_{(k)}(s^{(k)}_{h+1}, s^{(k)}_h, a^{(k)}_h)$ and $\widehat{o}^r_{k,h} = \widehat{r}^k_h(s^{(k)}_h, a^{(k)}_h)$.

It can be verified that the following relations hold at episode k for the state and action pairs from the k^{th} trajectory $\{s^{(k)}_0, a^{(k)}_0, s^{(k)}_1, a^{(k)}_1, .., s^{(k)}_{H-1}, a^{(k)}_{H-1}, s^{(k)}_H\}$:

$$P_{(k)}(s^{(k)}_{h+1} \mid s^{(k)}_h, a^{(k)}_h) = < \psi(s^{(k)}_{h+1}, s^{(k)}_h, a^{(k)}_h), o^p_{(k)}(s^{(k)}_{h+1}, s^{(k)}_h, a^{(k)}_h) > \quad, \forall h \in [H], \tag{D.3}$$

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

Note that the observed non-stationary parameters $\widehat{o}^{p}_{(k)}$ and $\widehat{o}^{r}_{(k)}$ can be interpreted partially observed vectors.

D.1.2 Functions f and g

The function f estimates and the function g predicts as follows:

1. **Function** f: f forecasts one-episode-ahead non-stationary parameters $\hat{o}^{p}_{(k+1)}$ and $\hat{o}^{r}_{(k+1)}$ by minimizing the following loss function $\mathcal{L}_{f^\diamond}$ with the 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 $\diamond = r$. We set $\square = \psi$ if $\diamond = p$. We let $\phi^k_{f^\diamond} = \text{argmin}_\phi \mathcal{L}_{f^\diamond}(\widehat{o}^{\diamond}_{k-(w-1):k})$. We use $\phi^k_{f^\diamond}$ as $\widehat{o}^{\diamond}_{k+1}$.

2. **Function** g: Then g predicts the functions $\widehat{P}_{(k+1)}$ and $\widehat{R}_{(k+1)}$, denoted as $\widehat{g}^{P}_{(k+1)}$ and $\widehat{g}^{R}_{(k+1)}$, 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 below:

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

The function $g \circ f$ forecasts the $k+1^{th}$ episode's non-stationary parameters 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 $o^{p}_{(k+1)}$ and $o^{r}_{(k+1)}$ by minimizing the following two regularized least-squares optimization problems [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. The function g predicts $\widehat{P}_{(k+1)}$ and $\widehat{R}_{(k+1)}$ from $\widehat{o}_{k+1}$.

From the equations (17a) and (17b) of the paper [31], the explicit solutions of (D.5) and (D.6) are given as

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

Then, the `ProST-T` predicts the future model using the functions $\widehat{g}^{P}_{k+1}$ and $\widehat{g}^{R}_{k+1}$ as follows:

$$\begin{aligned}
\widehat{g}^{P}_{k+1}(s',s,a) &\coloneqq < \varphi(s',s,a), \widehat{o}^{p}_{(k+1)}(s',s,a) >,\\
\widetilde{g}^{R}_{k+1}(s,a) &\coloneqq < \varphi(s,a), \widehat{o}^{r}_{(k+1)}(s,a) >,\\
\widehat{g}^{R}_{k+1}(s,a) &\coloneqq \widetilde{g}^{R}_{k+1}(s,a) + 2\Gamma^{(k)}_{w}(s,a).
\end{aligned}$$

We utilize the exploration bonus $\Gamma^{(k)}_{w}(s,a) : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ to explore those state and action pairs that are less visited. We define it as $\Gamma^{(k)}_{w}(s,a) = \beta\left(\sum_{t=k-w+1}^{k} n_t(s,a) + \lambda\right)^{-1/2}$ with $\beta > 0$. Then, we use $\widehat{g}^{P}_{k+1}$ and $\widehat{g}^{R}_{k+1}$ to denote the future MDP's $\widehat{P}_{(k+1)}$ and $\widehat{R}_{(k+1)}$, respectively. From the following analysis, we write $\widehat{P}_{(k+1)} = \widehat{g}^{P}_{(k+1)}$, $\widetilde{R}_{(k+1)} = \widetilde{g}^{R}_{(k+1)}$, and $\widehat{R}_{(k+1)} = \widehat{g}^{R}_{(k+1)}$.

D.1.3 Baseline algorithms `Alg` and $\texttt{Alg}_\tau$

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 $\pi^{(k)}$ is generated by the softmax transformation of $\theta^{(k)}$ at the wall-clock time $\mathfrak{t}_k$. Furthermore, to promote exploration and discourage premature convergence to suboptimal policies in a non-stationary environment, we implement 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 perform 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 $\pi^{(k)}$ is generated by the softmax transformation of $\theta^{(k)}$ at the wall-clock time $\mathfrak{t}_k$. Using the notation $\pi^{(k)} = \pi_{\theta^{(k)}}$, the soft parameterization is defined as

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

Under the softmax parameterization, the NPG update rule admits a simple form of update rule given in line 17 of Algorithm 2 in Appendix F.1. This is elaborated in [21].

Entropy regularized value maximization. For any policy π, we define the *forecasted* entropy-regularized state value function $\widehat{V}_\tau^{\pi, \mathfrak{t}_{\mathfrak{t}_k+1}}(s)$ as

$$\widehat{V}_\tau^{\pi, \mathfrak{t}_{k+1}}(s) \coloneqq \widehat{V}^{\pi, \mathfrak{t}_k+1}(s) + \tau\mathcal{H}(s, \pi)$$

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

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

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

$$\widehat{Q}_\tau^{\pi, \mathfrak{t}_{k+1}}(s,a) = \hat{r}_h^{\mathfrak{t}_{k+1}} + \gamma \mathbb{E}_{s' \sim \widehat{P}_{\mathfrak{t}_{k+1}}(\cdot|s,a)}\left[\widehat{V}_\tau^{\pi, \mathfrak{t}_{k+1}}(s')\right]$$
$$\text{where } (s', s, a) = (\hat{s}_{h+1}^{(k+1)}, \hat{s}_h^{(k+1)}, \hat{a}_h^{(k+1)}).$$

D.2 Notation for theoretical analysis

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

At the wall-clock time $\mathfrak{t}_k$, we define the *forecasting model error* $\Delta_{\mathfrak{t}_k}^r(s,a)$ and *forecasting transition probability model error* $\Delta_{\mathfrak{t}_k}^p(s,a)$ below:

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

Recall that $\widetilde{R}_{(k+1)}$ and $\widehat{P}_{(k+1)}$ estimate the future reward and transition probability by solving the optimization problems (D.5) and (D.6).

We define a model error that considers the bonus term as

$$\Delta_{\mathfrak{t}_k}^{Bonus,r}(s,a) \coloneqq \left|\left(R_{(k+1)} - \widehat{R}_{(k+1)}\right)(s,a)\right|$$

where $\widehat{R}_{(k+1)}(s,a) = \widetilde{R}_{(k+1)}(s,a) + 2\Gamma_w^{(k)}(s,a)$.

We also define the *empirical* forecasting reward model error $\bar{\Delta}_{\mathfrak{t}_k,h}^r$ and the *empirical* forecasting transition probability model error $\bar{\Delta}_{\mathfrak{t}_k,h}^p$:

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

as well as the *empirical* bonus based on the reward model error:

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

Likewise, we define *total empirical* forecasting reward model error $\bar{\Delta}^r_K$ and the *total empirical* forecasting transition probability model error $\bar{\Delta}^p_K$:

$$\bar{\Delta}^r_K \coloneqq \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \bar{\Delta}^r_{\mathfrak{t}_k,h}, \tag{D.10}$$

$$\bar{\Delta}^p_K \coloneqq \sum_{k=1}^{K-1}\sum_{h=0}^{H-1} \bar{\Delta}^p_{\mathfrak{t}_k,h}. \tag{D.11}$$

We simplify the symbols $\Delta^r_{\mathfrak{t}_k}(s,a), \Delta^p_{\mathfrak{t}_k}(s,a), \Delta^{Bonus,r}_{\mathfrak{t}_k}(s,a), \bar{\Delta}^r_{\mathfrak{t}_k,h}, \bar{\Delta}^p_{\mathfrak{t}_k,h}, \bar{\Delta}^{Bonus,r}_{\mathfrak{t}_k,h}$ as $\Delta^r_{(k)}(s,a), \Delta^p_{(k)}(s,a), \Delta^{Bonus,r}_{(k)}(s,a), \bar{\Delta}^r_{(k),h}, \bar{\Delta}^p_{(k),h}, \bar{\Delta}^{Bonus,r}_{(k),h}$, respectively.

We also define a variable $\Lambda^{\mathfrak{t}_k}_w(s,a)$ that quantifies the visitation:

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

It can be verified that

$$\Gamma^{\mathfrak{t}_k}_w(s,a) = \beta\sqrt{\Lambda^{\mathfrak{t}_k}_w(s,a)}. \tag{D.13}$$

As before, we simplify the notations $\Lambda^{\mathfrak{t}_k}_w(s,a)$ and $\Gamma^{\mathfrak{t}_k}_w(s,a)$ as $\Lambda^{(k)}_w(s,a)$ and $\Gamma^{(k)}_w(s,a)$. We define $r_{\max}, \widetilde{r}_{\max}, R^{\max}_{(k+1)}$, and $\widetilde{R}^{\max}_{(k+1)}$ as follows:

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

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

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

Also, since β and λ are hyperparameters independent of k, we have that

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

D.3 Proofs

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

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

1. Upper bound on 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 between two optimal value functions evaluated for two different MDPs: $\mathcal{M}_{(k+1)}$ and $\widehat{\mathcal{M}}_{(k+1)}$.

We will first come up with an upper bound on 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:

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

1.1. Terms (1.1) and (1.3).

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

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

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

1.2. Term (1.2).

An upper bound on the term (1.2) can be obtained by utilizing $\bar{\iota}_\infty^{(k+1)}(s,a)$ (Def (C.10)). Then, the Q-function gap between $Q_\infty^{*,(k+1)}$ and $\widehat{Q}_\infty^{*,(k+1)}$ can be represented using the Bellman equation as follows:

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

where (D.16) and (D.17) hold by the definition of Bellman equation ((C.3) and (C.7)). Equation (D.18) holds by $\langle \widehat{Q}_\infty^{*,(k+1)}, \pi^{*,(k+1)} - \widehat{\pi}^{*,(k+1)} \rangle_{\mathcal{A}}(s,a) \leq 0$ 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 captures how the state-action pair transitions from (s,a) to (s',a') when following the policy π in an environment with the transition probability $\mathbb{P}$. Also, define the one-vector $\mathbb{1}_{(s,a)} \in \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|}$ such that the $(s,a)^{\text{th}}$ entity is one and the remaining entries are zero. Then, the equation (D.15) becomes 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 right-hand side of equation (D.18) can be represented as

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

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

Then, we can reformulate the inequality (between (D.15) and (D.18)) into a vector form which holds element-wise for all s,a:

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

Then, rearranging the above inequality yields that

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

Now, note that $(\mathbb{I} - \gamma\mathbb{P}_{\pi*}^{(k+1)})^{-1}$ can be expanded with an infinite summation of the matrix operator $P_{(k+1)} \circ \pi^{*,(k+1)}$ as $(\mathbb{I} - \gamma\mathbb{P}_{\pi*}^{(k+1)})^{-1} = \mathbb{I} + \gamma\mathbb{P}_{\pi*}^{(k+1)} + (\gamma\mathbb{P}_{\pi*}^{(k+1)})^2 + ...$s. Since, $\mathbb{1}_{(s,a)}$ can be viewed as the Dirac delta state-action distribution that always yields (s,a), it holds that $\nu_{(s,a)}^{\pi^{*,(k+1)},(k+1)} = (\mathbb{I} - \gamma\mathbb{P}_{\pi*}^{(k+1)})^{-1}\mathbb{1}_{(s,a)}$, where ν is the unnormalized occupancy measure of (s,a) in light of Definition (C.8). Then taking the l_1 norm over the inequality (D.19) yields the that

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

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

Then, for every $(s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$, it follows from combining the terms (1.1), (1.2) and (1.3) that

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

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

Finally, an upper bound on ① is derived as

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

where we have defined $\bar{\iota}_\infty^K \coloneqq \sum_{k=1}^{K-1} \left\| \bar{l}_\infty^{(k+1)} \right\|_\infty$ in Theorem 1.

2. Upper bound on ②.

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

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

where the subscript ∞ in the notations $\widehat{V}_\infty^{\pi,(k+1)}(s_0)$ and $\widehat{V}_{\infty,\tau}^{\pi,(k+1)}(s_0)$ indicate the forecasted value function and the 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 all $\pi \in \Pi$.

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

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

due to Theorem 5.3 in [38]. Now, combining D.23 and D.24 offers an upper bound of the term ②'s $(k)^{\text{th}}$ term as follows:

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

Hence,

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

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

The term (2.1) in (D.23) can be further bounded as follows:

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

where (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 all π.

We now bound the term (2.2) in (D.26). With the policy-update rule of `ProST-T` (Algorithm 2 in Appendix F.2), suppose that for a given $g \in [\Delta_\pi]$, we have obtained an *inexact* soft Q-function value of the policy $\widehat{\pi}_{(g)}$ as $\widetilde{Q}_{\tau}^{\widehat{\pi}_{(g)}}$, where $\widehat{Q}_{\tau}^{\widehat{\pi}_{(g)}}$ denotes an *exact* soft forecated Q-function value and g is the iteration index. The approximation gap $|\widetilde{Q}_{\tau}^{\widehat{\pi}_{(g)}} - \widehat{Q}_{\tau}^{\widehat{\pi}_{(g)}}|$ results from computing Q using a finite number of samples. For a hyperparameter δ, let the maximum of the approximation gap over (s,a) is smaller than δ, namely $\|\widetilde{Q}_{\tau}^{\widehat{\pi}_{(g)}} - \widehat{Q}_{\tau}^{\widehat{\pi}_{(g)}}\|_{\infty} \leq \delta$ holds. Then, for iteration $g = 1, 2, .., \Delta_\pi$, the policy-update rule of `ProST-T` can be written as

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

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

In light of Theorem 2 in [21], when the learning rate is such that $0 \leq \eta \leq (1-\gamma)/\tau$, then the approximate entropy-regularized NPG method satisfies the linear convergence theorem for every $g \in [\Delta_\pi]$:

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

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

where

$$
\begin{aligned}
C_1 &:= \|\widehat{Q}_{\tau}^{*,(k+1)} - \widehat{Q}_{\tau}^{\widehat{\pi}_{(0)}}\|_{\infty} + 2\tau\left(1 - \frac{\eta\tau}{1-\gamma}\right)\|\log \widehat{\pi}^{*,(k+1)} - \log \widehat{\pi}_{(0)}\|_{\infty} \\
&= \|\widehat{Q}_{\tau}^{*,(k+1)} - \widehat{Q}_{\tau}^{\pi^{(k)}}\|_{\infty} + 2\tau\left(1 - \frac{\eta\tau}{1-\gamma}\right)\|\log \widehat{\pi}^{*,(k+1)} - \log \widehat{\pi}^{(k)}\|_{\infty} \qquad \text{(D.29)} \\
C_2 &:= \frac{2\delta}{1-\gamma}\left(1 + \frac{\gamma}{\eta\tau}\right) \qquad \text{(D.30)}
\end{aligned}
$$

The equation (D.29) holds since the policy that the agent executes at the wall-clock time $\mathfrak{t}_k$ (episode k), i.e., $\pi^{(k)}$, is same as the initial policy of the policy iteration, i.e., $\widehat{\pi}_{(0)}$, at the wall-clock time $\mathfrak{t}_k$. Also, the policy that the agent executes at the wall-clock time $\mathfrak{t}_{k+1}$, i.e., $\widehat{\pi}^{(k+1)}$, is same as the policy after Δ_π steps of the soft policy iteration, i.e., $\widehat{\pi}_{(\Delta_\pi)}$ at the wall-clock time $\mathfrak{t}_{k+1}$.

Now, the term (2.2) can be bounded as follows:

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

Combining (D.23,D.26 and D.31) offers an upper bound on the term ②'s $k^{(th)}$ term as follows,

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

Hence,

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

where (D.32) and (D.33) hold when $0 \le \eta \le (1-\gamma)/\tau$

3. Upper bound on ③.

By recalling Definition (C.11), note that $\iota_h^{(k+1)}(\widehat{s}_h^{(k+1)}, \widehat{a}_h^{(k+1)})$ is an *empirical* estimated model prediction error, measuring the gap between $\mathcal{M}_{(k+1)}$ and $\widehat{\mathcal{M}}_{(k+1)}$. Specifically, at episode k, the `ProST` algorithm creates the future MDP $\widehat{\mathcal{M}}_{(k+1)}$ and evaluates $\widehat{V}$ and $\widehat{Q}$ using $\widehat{\pi}^{(k+1)}$. Subsequently at episode $k+1$, the agent uses $\widehat{\pi}^{(k+1)}$ to rollout a trajectory $\{s_0^{(k+1)}, a_0^{(k+1)}, s_1^{(k+1)}, a_1^{(k+1)}, ..., s_{H-1}^{(k+1)}, a_{H-1}^{(k+1)}, s_H^{(k+1)}\}$. Based on this observation, one can write

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

Equation (D.34) holds due to (C.6). Now, we define the operator $\widehat{\mathfrak{T}}^{(k+1)}$ for a function $f : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ as follows:

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

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 $V_h^{\widehat{\pi}^{(k+1)},(k+1)}(s) = \left\langle Q_h^{\widehat{\pi}^{(k+1)},(k+1)}, \widehat{\pi}^{(k+1)}\right\rangle_{\mathcal{A}}$ in light of (C.6) and (C.2). 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

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

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

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

Therefore, we have the following recursive formula over h:

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

The summation over $h = 0,1,..,H-1$ yields that

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

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 a 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 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 estimates $\widehat{V}_h^{\pi,(k+1)}$ and $\widehat{Q}_h^{\pi,(k+1)}$ are $\mathcal{F}_{1,1}^{(k+1)}$ measurable since they are forecasted from the past k historical trajectories. Now, since $D_{h,1}^{(k+1)} \in \mathcal{F}_{h,1}^{(k+1)}$ and $D_{h,2}^{(k+1)} \in \mathcal{F}_{h,2}^{(k+1)}$ hold, $\mathbb{E}[D_{h,1}^{(k+1)}|\mathcal{F}_{h-1,2}^{(k+1)}] = 0$ and $\mathbb{E}[D_{h,2}^{(k+1)}|\mathcal{F}_{h,1}^{(k+1)}] = 0$. 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, one can define a martingale sequence adapted to the filtration $\{\mathcal{F}_{h,m}^{(k)}\}_{(k,h,m)\in[K]\times[H]\times[2]}$:

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

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

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

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 $1 - p/2$. The term ③ can be bounded as

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

4. Upper bound on dynamic regret.

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

For without entropy-regularized case, combining the equations (D.21), (D.25) and (D.35) leads to the following upper bound on the dynamic regret for a future policy $\{\widehat{\pi}\}$ that holds with probability at least $1 - p/2$:

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

Taking an upper bound on $r_{\max}$ and $\hat{r}_{\max}$ using $(r_{\max} \vee \hat{r}_{\max})$ yields the following upper bound that holds with probability at least $1 - p/2$:

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

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

For the entropy-regularized case, combining the equations (D.21), (D.33), (D.35) leads to the following upper bound on the dynamic regret for a future policy $\{\widehat{\pi}\}$ that holds with probability at least $1 - p/2$:

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

Then, the following holds with probability at least $1-p/2$:

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

4.3. Upper bound of Theorem 1.

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

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

where $\mathfrak{R}_{II} = \mathfrak{R}_{\texttt{Alg}}$ if we use $\texttt{Alg}$ as the baseline algorithm and $\mathfrak{R}_{II} = \texttt{Alg}_\tau$ if we use $\mathfrak{R}_{\texttt{Alg}_\tau}$ as the baseline algorithm:

$$
\begin{aligned}
\mathfrak{R}_I &= \frac{1}{1-\gamma}\bar{\iota}_{\infty}^{K} - \iota_H^{(k)} + C_p\sqrt{K-1} \\
\mathfrak{R}_{\texttt{Alg}} &= C_{\texttt{Alg}}(\Delta_\pi)\cdot(K-1) \\
\mathfrak{R}_{\texttt{Alg}_\tau} &= C_{\texttt{Alg}_\tau}(\Delta_\pi)\cdot(K-1)
\end{aligned}
$$

where the 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 and H to guarantee the optimal threshold 2ϵ of ② without entropy regularization**). *We decompose the term ② as*

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

To guarantee that the terms ② − @ and ② − ⓑ are each less than or equal to ϵ, it suffices to satisfy the following conditions for τ, η, Δ_π and H:

$$
\begin{aligned}
&②-@ : \Delta_\pi \geq \left(\frac{1}{(1-\gamma)^2} + \frac{\log|\mathcal{A}|}{\eta}\right)\cdot\frac{1}{\epsilon} \\
&②-ⓑ : H \geq \frac{\log(\frac{1-\gamma}{2\widehat{r}_{max}}\epsilon)}{\log(\gamma)} \quad \textit{or} \quad H \geq \frac{1}{1-\gamma}\log\left(\frac{2\widehat{r}_{max}}{(1-\gamma)\epsilon}\right)
\end{aligned}
$$

Lemma 2 (Conditions on τ, Δ_π, H to guarantee the optimal threshold 4ϵ of ② with entropy regularization). *We decompose the term ② as*

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

To guarantee that the terms ② − ⓑ, ② − ⓒ and ② − ⓓ are each less than or equal to ϵ, it suffices to satisfy the following conditions for τ, η, Δ_π and H:

$$②-ⓑ : \delta \le \frac{\epsilon}{(\gamma+2)\cdot\frac{2}{1-\gamma}\cdot(1+\frac{\gamma}{\eta\tau})} \tag{D.36}$$

$$②-ⓒ : H \ge \frac{\log(\frac{1-\gamma}{2\widehat{r}_{max}}\epsilon)}{\log(\gamma)} \quad or \quad H \ge \frac{1}{1-\gamma}\log\left(\frac{2\widehat{r}_{max}}{(1-\gamma)\epsilon}\right) \tag{D.37}$$

$$②-ⓓ : \tau \le \frac{1-\gamma}{2\log|\mathcal{A}|}\epsilon \tag{D.38}$$

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

$$②-ⓐ : \Delta_\pi \ge \frac{\log\left(\frac{\epsilon}{C_1(\gamma+2)}\right)}{\log(1-\eta\tau)} + 1 \quad or \quad \Delta_\pi \ge \frac{1}{\eta\tau}\log\left(\frac{C_1(\gamma+2)}{\epsilon}\right) + 1 \tag{D.39}$$

The inequalities (D.37) and (D.39) results from applying the first-order Taylor series on $\log(\gamma)$ and $\log(1-\eta\tau)$ since $\gamma \in (0,1]$ and $\eta \in (0,(1-\gamma)/\tau]$. The inequalities (D.36) and (D.39) implies that if the learning rate η is fixed in the admissible range, then the iteration complexity scales inversely proportional to τ, and the upper bound on δ, which we will denote it as δ_{max}, also scales proportional to τ.

Now, the best guaranteed convergence can be achieved when $\eta^* = (1-\gamma)/\tau$ (associated with the value of η that minimizes the equation (D.29)), for which conditions of hyperparameters Δ_{π,η^*} and δ_{η^*} are

$$②-ⓐ : \Delta_{\pi,\eta^*} \ge \frac{1}{1-\gamma}\log\left(\frac{\|\widehat{Q}_\tau^{*,(k+1)} - \widehat{Q}_\tau^{\widehat{\pi}^{(0)}}\|_\infty(\gamma+2)}{\epsilon}\right) + 1$$

$$②-ⓑ : \delta_{\eta^*} \le \frac{\epsilon(1-\gamma)^2}{2(\gamma+2)}.$$

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

Lemma 3 (Sample complexity to guarantee the optimal threshold 4ϵ of ②). *We define δ_{max} as right-hand side of the equation (D.36). If we have the number of samples per state-action pairs is at least the order of*

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

up to some logarithmic factor, then $\delta \le \delta_{max}$ holds with high probability and we can guarantee the optimal threshold 4ϵ with high probability for the upper bound of ②, provided (D.37), (D.38) and (D.39) hold.

Proof of Theorem 2. **1. ProST-T $\iota_H^{(k)}$:**

The *empirical* estimated model prediction error $\iota_h^{(k+1)}(s_h^{(k+1)}, a_h^{(k+1)})$ is represented as follows (Definition (C.11)):

$$
\begin{aligned}
-\iota_h^{(k+1)}(s_h^{(k+1)}, a_h^{(k+1)}) =& -R_{(k+1)}(s_h^{(k+1)}, a_h^{(k+1)}) - \gamma(P_{(k+1)}\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)})(s_h^{(k+1)}, a_h^{(k+1)}) \\
&+ \widehat{Q}_h^{\widehat{\pi}^{(k+1)},(k+1)}(s_h^{(k+1)}, a_h^{(k+1)}) && \text{(D.40)} \\
=& -R_{(k+1)}(s_h^{(k+1)}, a_h^{(k+1)}) - \gamma(P_{(k+1)}\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)})(s_h^{(k+1)}, a_h^{(k+1)}) \\
&+ \widehat{R}_{(k+1)}(s_h^{(k+1)}, a_h^{(k+1)}) + \gamma(\widehat{P}_{(k+1)}\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)})(s_h^{(k+1)}, a_h^{(k+1)}) && \text{(D.41)} \\
=& \left(\widehat{R}_{(k+1)} - R_{(k+1)}\right)(s_h^{(k+1)}, a_h^{(k+1)}) \\
&+ \gamma\left(\left(\widehat{P}_{(k+1)} - P_{(k+1)}\right)\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)}\right)(s_h^{(k+1)}, a_h^{(k+1)}) \\
\leq& \bar{\Delta}_{(k),h}^{Bonus,r} + \gamma\left\|\left(\widehat{P}_{(k+1)} - P_{(k+1)}\right)(\cdot \mid s_h^{(k+1)}, a_h^{(k+1)})\right\|_1 \left\|\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)}(\cdot)\right\|_\infty \\
\leq& \bar{\Delta}_{(k),h}^{Bonus,r} + \gamma\bar{\Delta}_{k,h}^{p}\frac{\gamma^{H-h}\hat{r}_{\max}}{1-\gamma} && \text{(D.42)} \\
\leq& \bar{\Delta}_{(k),h}^{r} + 2\Gamma_w^{(k)}(s_h^{(k+1)}, a_h^{(k+1)}) + \gamma\bar{\Delta}_{(k),h}^{p}\frac{\gamma^{H-h}\hat{r}_{\max}}{1-\gamma} && \text{(D.43)}
\end{aligned}
$$

The equation (D.41) holds due to the future Bellman equation (C.6), the equation (D.42) holds since $\left\|\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)}(\cdot)\right\|_\infty \leq \sum_{h'=h+1}^{H}\gamma^{h'-(h+1)}\hat{r}_{\max} \leq \gamma^{H-h}\hat{r}_{\max}/(1-\gamma)$, and the equation (D.43) holds since $\Delta_{(k)}^{Bonus,r}(s,a) \leq \left|\left(R_{(k+1)} - \widetilde{R}_{(k+1)}\right)(s,a)\right| + |2\Gamma_w^{(k)}(s,a)| = \Delta_{(k)}^{r}(s,a) + 2\Gamma_w^{(k)}(s,a)$ for all (s,a). The summation of the empirical model prediction error over all episodes and all steps can be bounded as

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

We use Lemma 8 to bound the term ①, Lemma 9 and (D.13) to bound the term ②, and Lemma 11 (or Lemma 10) to bound the term ③:

$$
\begin{aligned}
① &\leq wHB_r(\Delta_\pi) + \lambda r_{\max}\cdot(K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} && \text{(D.45)} \\
② &\leq 2\beta(K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} && \text{(D.46)} \\
③ &\leq \left(|\mathcal{S}|\sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda\right)(K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} + wHB_p(\Delta_\pi) && \text{(D.47)}
\end{aligned}
$$

where the inequality (D.47) holds with probability at least $1-\delta$, where $\delta \in (0,1)$. Now, combining (D.45), (D.46) and (D.47) that

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

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

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

$$
\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)} \\
\leq& |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} \\
\leq& \left(B_r^{(k-w+1: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^{(k-w+1: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)} \\
\leq& \left(B_r^{(k-w+1: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^{(k-w+1: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)} \\
\leq& B_r^{(k-w+1:k)}(\Delta_\pi) + \gamma B_p^{(k-w+1: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)}_{\leq 0}(\Lambda_w^{(k)}(s,a))^{1/2} && \text{(D.57)} \\
\leq& B_r^{(k-w+1:k)}(\Delta_\pi) + \gamma B_p^{(k-w+1:k)}(\Delta_\pi) && \text{(D.58)}
\end{aligned}
$$

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

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

or equivalently,

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

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

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

because $\sum_{k=1}^{K-1} B_p^{(k-w+1:k)}(\Delta_\pi) = \sum_{\mathcal{E}=1}^{\lfloor \frac{K-1}{w} \rfloor} \sum_{k=(\mathcal{E}-1)w}^{\mathcal{E}w} B_p^{(k-w+1:k)}(\Delta_\pi) \le w B_p(\Delta_\pi)$ holds and in the same way $\sum_{k=1}^{K-1} B_p^{(k-w+1:k)}(\Delta_\pi) \le w B_r(\Delta_\pi)$ holds.

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

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

Finally, the term $\mathfrak{R}_I$ can be bounded as

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

Now, let $B(\Delta_\pi)$ be a conic combination of $B_r(\Delta_\pi)$ and $B_p(\Delta_\pi)$ as

$$\begin{aligned}
B(\Delta_\pi) &= \left(\frac{1}{1-\gamma} + H\right)B_r(\Delta_\pi) + \frac{(1+H\hat{r}_{\max})\gamma}{1-\gamma}B_p(\Delta_\pi) \\
&\le \left(\frac{1}{1-\gamma} + H\right)\Delta_\pi^{\alpha_r}B_r(1) + \frac{(1+H\hat{r}_{\max})\gamma}{1-\gamma}\Delta_\pi^{\alpha_p}B_p(1) \\
&= C_{B_r}\Delta_\pi^{\alpha_r} + C_{B_p}\Delta_\pi^{\alpha_p}
\end{aligned} \tag{D.60}$$

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

Recall the definitions of $B_r(\Delta_\pi)$ and $B_p(\Delta_\pi)$, as well as the inequalities $B_r(\Delta_\pi) \leq \Delta_\pi^{\alpha_r} B_r(1)$ and $B_p(\Delta_\pi) \leq \Delta_\pi^{\alpha_p} B_p(1)$. We denote $B_p(1)$ and $B_r(1)$ as time-elapsing variation budgets for one policy iteration. We also let the constant C_k be defined as

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

Then, an upper bound on $\mathfrak{R}_I$ can be obtained as

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

□

Proof of Proposition 2. Now, we set the sliding window length w that is adaptive to Δ_π as follows:

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

Then,

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

Since C_k is linear to $K-1$, the function $\mathfrak{R}_I$ satisfies that

$$\mathfrak{R}_I = \mathcal{O}\left(B(\Delta_\pi)^{1/3}(K-1)^{2/3} \cdot \sqrt{\log\left(\frac{K-1}{B(\Delta_\pi)}\right)}\right). \tag{D.61}$$

Now, by utilizing (D.60), if $B(\Delta_\pi) \leq C_{B_r}\Delta_\pi^{\alpha_r} + C_{B_p}\Delta_\pi^{\alpha_p} = o(K)$ holds, then $\mathfrak{R}_I$ is sublinear to K. The corresponding condition is $B_r(1) + \frac{\hat{r}_{\max}}{1-\gamma}B_p(1) = o(K)$ with $\Delta_\pi < K$ since

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

This completes the proof. □

Proof of Theorem 3. We first prove multiple statements below. We denote the upper bound on $\mathfrak{R}_I$ as $\mathfrak{R}_I^{\max}$, and that of $\mathfrak{R}_{II}$ as $\mathfrak{R}_{II}^{\max}$.

1. The upper bound on $\mathfrak{R}_{II}(\Delta_\pi)$ (i.e., $\mathfrak{R}_{II}^{\max}$) is a non-increasing function, the upper bound on $\mathfrak{R}_I(\Delta_\pi)$ (i.e., $\mathfrak{R}_I^{\max}$) is a non-decreasing function , and both are convex in the region $\Delta_\pi \in$

$\mathbb{N}_I \cap \mathbb{N}_{II}$

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

since $\Delta_\pi \in \mathbb{N}_I \cap \mathbb{N}_{II}$ satisfies $\Delta_\pi > 1$ and $\log(1-\eta\tau) \leq 0$ holds under the hyperparameter assumption $0 \leq \eta \leq (1-\gamma)/\tau$, it follows from the Proposition 1 that

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

when $\alpha_r, \alpha_p \geq 1$.

2. Suboptimal Δ_π^*

We slightly relax the upper bound $\mathfrak{R}_I(\Delta_\pi) \leq C_{B_r}\Delta_\pi^{\alpha_r} + C_{B_p}\Delta_\pi^{\alpha_p}$ to $\mathfrak{R}_I(\Delta_\pi) = \left(C_{B_r} + C_{B_p}\right)\Delta_\pi^{\max(\alpha_r,\alpha_p)}$ and obtain Δ_π^* in the worst case by optimizing $\mathfrak{R}_I^{\max}(\Delta_\pi) + \mathfrak{R}_{II}^{\max}(\Delta_\pi)$.

1. $\boldsymbol{\max(\alpha_r, \alpha_p) = 0}$: this means that $\mathfrak{R}_I^{\max}(\Delta_\pi) = C_{B_r} + C_{B_p}$, where $\mathfrak{R}_I^{\max}$ is now independent of Δ_π. Then, an infinite number Δ_π guarantees a small dynamic regret $\mathfrak{R}_I$, which also leads to a small $\mathfrak{R}$. It can be checked that $\mathfrak{R}_{II}$ without entropy regularization decreases with the scale of $1/\Delta_\pi$, and $\mathfrak{R}_{II}$ with entropy regularization decreases with the scale of $\exp(\Delta_\pi)$. This also matches with the existing results on achieving a faster convergence with an entropy regularization.

For the remaining case, we first compute the gradient of the term $\mathfrak{R}_I^{\max}(\Delta_\pi) + \mathfrak{R}_{II}^{\max}(\Delta_\pi)$ when $\mathfrak{R}_{II}^{\max}(\Delta_\pi)$ comes from entropy-regularized case:

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

when $\mathfrak{R}_{II}^{\max}(\Delta_\pi)$ is for the case without entropy regularization, the gradient of the dynamic regret upper bound is given as

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

2. $\boldsymbol{\max(\alpha_r, \alpha_p) = 1}$: The relation $(1-\eta\tau)^{\Delta_\pi - 1} = k_I/k_{II}$ should be satisfied for the entropy regularized case and $\Delta_\pi^{-2} = k_I/k_{II}$ should be satisfied in the case without entropy regularization, respectively. Then, it holds that $\Delta_\pi^* = \log_{1-\eta\tau}(k_I/k_{II}) + 1$ for the entropy regularized case and $\Delta_\pi^* = \sqrt{k_{II}/k_I}$ without regularization.

Now, for the case of the entropy regularized case, if $k_{II} = (1-\eta\tau)k_I$ is satisfied, $\partial\left(\mathfrak{R}_I^{\max}(\Delta_\pi)+\mathfrak{R}_{II}^{\max}(\Delta_\pi)\right)/\partial\Delta_\pi = 0$ is equal to solving $\Delta_\pi^{\max(\alpha_r,\alpha_p)-1} = (1-\eta\tau)^{\Delta_\pi}$. Now, we use the Lambert W function to find Δ_π as follows:

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

3. $\mathbf{0 < \max(\alpha_r,\alpha_p) < 1}$:
   - Without Entropy-regularization: $\Delta_\pi^* = (k_I/k_{II})^{1/(\max(\alpha_r,\alpha_p)+1)}$
   - With Entropy-regularization: Since $x = -\frac{\log(1-\eta\tau)}{\max(\alpha_r,\alpha_p)-1} < 0$, a small $|x|$ will have a large $-W(x) > 0$ value, which leads to a large Δ_π^*.
4. $\mathbf{\max(\alpha_r,\alpha_p) > 1}$:
   - Without Entropy-regularization: $\Delta_\pi^* = (k_I/k_{II})^{1/(\max(\alpha_r,\alpha_p)+1)}$
   - With Entropy-regularization: It holds that $x > 0$ and $-W(x) < 0$. Then $\Delta_\pi^* < 1$, which means that one iteration is enough.

□

From the proof of Theorem 2, we will develop Lemma 4, Lemma 5 and Lemma 6 to upper-bound two model prediction errors $-\iota_h^{(k)}$ and $\bar{\iota}_\infty^k$.

Lemma 4 (Upper bound on $-\iota_h^{(k+1)}(s_h^{(k+1)}, a_h^{(k+1)})$ by $\bar{\Delta}_{k,h}^r,\ \bar{\Delta}_{k,h}^p$). *It holds that*

$$
-\iota_h^{(k+1)}(s_h^{(k+1)}, a_h^{(k+1)}) \le \bar{\Delta}_{k,h}^r + 2\Gamma_w^{(k)}(s,a) + \gamma\bar{\Delta}_{k,h}^p\frac{\gamma^{H-h}\hat{r}_{max}}{1-\gamma}
$$

Proof of Lemma 4. It follows from (D.40), (D.41), (D.42) and (D.43). □

Lemma 5 (Upper bound on $-\iota_h^{(k+1)}(s,a)$ by $\Delta_{(k)}^r,\ \Delta_{(k)}^p$). *For every $(s,a) \in \mathcal{S}\times\mathcal{A}$, it holds that*

$$
-\iota_h^{(k+1)}(s,a) \le \Delta_{(k)}^r(s,a) + \gamma\Delta_{(k)}^p(s,a)\frac{\gamma^{H-h}\hat{r}_{max}}{1-\gamma} + 2\Gamma_w^{(k)}(s,a)
$$

Proof of Lemma 5.

$$
\begin{aligned}
-\iota_h^{(k+1)}(s,a) =& -R_{(k+1)}(s,a) - \gamma(P_{(k+1)}\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)})(s,a) + \widehat{Q}_h^{\widehat{\pi}^{(k+1)},(k+1)}(s,a) \\
=& -R_{(k+1)}(s,a) - \gamma(P_{(k+1)}\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)})(s,a) \\
&+ \widehat{R}_{(k+1)}(s,a) + \gamma(\widehat{P}_{(k+1)}\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)})(s,a) \\
=& \left(\widehat{R}_{(k+1)} - R_{(k+1)}\right)(s,a) + \gamma\left(\left(\widehat{P}_{(k+1)} - P_{(k+1)}\right)\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)}\right)(s,a) \\
\leq& \Delta_{(k)}^r(s,a) + 2\Gamma_w^{(k)}(s,a) + \gamma\left\|\left(\widehat{P}_{(k+1)} - P_{(k+1)}\right)(\cdot \mid s,a)\right\|_1 \left\|\widehat{V}_{h+1}^{\widehat{\pi}^{(k+1)},(k+1)}(\cdot)\right\|_\infty \\
\leq& \Delta_{(k)}^r(s,a) + 2\Gamma_w^{(k)}(s,a) + \gamma\Delta_{(k)}^p(s,a)\frac{\gamma^{H-h}\hat{r}_{\max}}{1-\gamma}
\end{aligned}
$$

□

Lemma 6 (Upper bound on $\bar{\iota}_\infty^k$ by $\Delta_{(k)}^r,\ \Delta_{(k)}^p$)**.** *For every* $(s,a) \in \mathcal{S} \times \mathcal{A}$*, it holds that*

$$
\bar{\iota}_\infty^{k+1}(s,a) \leq \Delta_{(k)}^r(s,a) + \Delta_{(k)}^p(s,a)\frac{\gamma\hat{r}_{max}}{1-\gamma} - 2\Gamma_w^{(k)}(s,a)
$$

Proof of Lemma 6. It results from (D.52),

$$
\begin{aligned}
\bar{\iota}_\infty^{k+1} &= \left(R_{(k+1)} - \widetilde{R}_{(k+1)}\right) - \beta(\Lambda_w^{(k)}(s,a))^{1/2} + \gamma\left(P_{(k+1)} - \widehat{P}_{(k+1)}\right)\widehat{V}_\infty^{*,(k+1)} - \beta(\Lambda_w^{(k)}(s,a))^{1/2} \\
&\leq \left|R_{(k+1)} - \widetilde{R}_{(k+1)}\right| - \beta(\Lambda_w^{(k)}(s,a))^{1/2} + \gamma\left\|P_{(k+1)} - \widehat{P}_{(k+1)}\right\|_1 \left\|\widehat{V}_\infty^{*,(k+1)}\right\|_\infty - \beta(\Lambda_w^{(k)}(s,a))^{1/2} \\
&\leq \Delta_{(k)}^r(s,a) - \beta(\Lambda_w^{(k)}(s,a))^{1/2} + \gamma\Delta_{(k)}^p(s,a)\frac{\hat{r}_{max}}{1-\gamma} - \beta(\Lambda_w^{(k)}(s,a))^{1/2} \\
&= \Delta_{(k)}^r(s,a) + \Delta_{(k)}^p(s,a)\frac{\gamma\hat{r}_{max}}{1-\gamma} - 2\Gamma_w^{(k)}(s,a)
\end{aligned}
$$

□

Lemma 7 (Upper bound on $\Delta_{(k)}^r(s,a)$)**.** *For every* $(s,a) \in \mathcal{S} \times \mathcal{A}$*, it holds that*

$$
\Delta_{(k)}^r(s,a) \leq B_r^{(k-w:k)}(\Delta_\pi) + \lambda\Lambda_w^{(k)}(s,a)r_{max}
$$

Proof of Lemma 7. We directly utilize the proof of Lemma 35 in [31]. For every $(s,a) \in \mathcal{S} \times \mathcal{A}$, $\Delta^r_{(k)}(s,a)$ can be represented as

$$\Delta^r_{(k)}(s,a) \tag{D.62}$$

$$= |R_{(k+1)}(s,a) - \widetilde{R}_{(k+1)}(s,a)| \tag{D.63}$$

$$= |o^r_{(k+1)}(s,a) - \widetilde{o}^r_{(k+1)}(s,a)| \tag{D.64}$$

$$= \left| \frac{\sum_{t=(1\wedge k-w+1)}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot r_h^t}{\lambda + \sum_{t=(1\wedge k-w+1)}^{k} n_t(s,a)} - o^r_{(k+1)}(s,a) \right| \tag{D.65}$$

$$= \Lambda^{(k)}_w(s,a) \left| \sum_{t=(1\wedge k-w+1)}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot r_h^t - \left(\lambda + \sum_{t=(1\wedge k-w+1)}^{k} n_t(s,a)\right) o^r_{(k+1)}(s,a) \right| \tag{D.66}$$

$$= \Lambda^{(k)}_w(s,a) \left| \sum_{t=(1\wedge k-w+1)}^{k} \sum_{h=0}^{H-1} \left(\mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \left(r_h^t - o^r_{(k+1)}(s,a)\right)\right) - \lambda \cdot o^r_{(k+1)}(s,a) \right| \tag{D.67}$$

$$\leq \Lambda^{(k)}_w(s,a) \left(\sum_{t=(1\wedge k-w+1)}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot \left| r_h^t - o^r_{(k+1)}(s,a) \right| \right) + \lambda \Lambda^{(k)}_w(s,a) \left| o^r_{(k+1)}(s,a) \right| \tag{D.68}$$

$$\leq \Lambda^{(k)}_w(s,a) \left(\sum_{t=(1\wedge k-w+1)}^{k} n_t(s,a) \left(\left| r^t(s,a) - o^r_{(k+1)}(s,a) \right| \right) \right) + \lambda \Lambda^{(k)}_w(s,a) r_{\max} \tag{D.69}$$

$$\leq \max_{(1\wedge k-w+1) \leq t \leq k} \left(\left| r^t(s,a) - o^r_{(k+1)}(s,a) \right| \right) \Lambda^{(k)}_w(s,a) \left(\sum_{t=(1\wedge k-w+1)}^{k} n_t(s,a) \right) + \lambda \Lambda^{(k)}_w(s,a) r_{\max}$$

$$\leq \max_{(1\wedge k-w+1) \leq t \leq k} \left(\left| r^t(s,a) - o^r_{(k+1)}(s,a) \right| \right) + \lambda \Lambda^{(k)}_w(s,a) r_{\max}$$

$$\leq B_r^{(k-w:k)}(\Delta_\pi) + \lambda \Lambda^{(k)}_w(s,a) r_{\max} \tag{D.70}$$

Equations (D.64) and (D.65) hold by the definition of $o^r_{k+1}, \widetilde{o}^r_{k+1}$ (definition (D.7)), equation (D.66) holds by the definition (D.12), equation (D.67) holds since $n_t(s,a) \coloneqq \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right]$, and inequality (D.70) holds since $\max_{(1\wedge k-w+1) \leq t \leq k} \left(\left| r^t(s,a) - o^r_{(k+1)}(s,a) \right| \right) \leq |r^{(1\wedge k-w+1)}(s,a) - r^{(1\wedge k-w+1)+1}(s,a)| + \cdots + |r^k(s,a) - r^{k+1}(s,a)| = B_r^{(k-w:k)}(\Delta_\pi)$. □

Lemma 8 (Upper bound on $\bar{\Delta}^r_K$)**.** *For every $(s,a) \in \mathcal{S} \times \mathcal{A}$, it holds that*

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

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

$$\begin{aligned}
\bar{\Delta}_K^r &= \sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\bar{\Delta}_{k,h}^r \\
&= \sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\Delta_{(k)}^r(s_h^{(k+1)}, a_h^{(k+1)}) \\
&\le \sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\left(B_r^{(k-w:k)}(\Delta_\pi) + \lambda\Lambda_w^{(k)}(s_h^{(k+1)}, a_h^{(k+1)})r_{\max}\right) \qquad \text{(D.71)} \\
&= wHB_r(\Delta_\pi) + \lambda r_{\max}\cdot\sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\left(\Lambda_w^{(k)}(s_h^{(k+1)}, a_h^{(k+1)})\right) \qquad \text{(D.72)} \\
&\le wHB_r(\Delta_\pi) + \lambda r_{\max}\cdot\sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\left(\sqrt{\Lambda_w^{(k)}(s_h^{(k+1)}, a_h^{(k+1)})}\right) \\
&\le wHB_r(\Delta_\pi) + \lambda r_{\max}\cdot(K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} \qquad \text{(D.73)}
\end{aligned}$$

The inequality (D.71) holds by Lemma 7, the equation (D.72) holds since $\sum_{k=1}^{K-1} B_r^{(k-w:k)}(\Delta_\pi) = \sum_{\mathcal{E}=1}^{\lfloor\frac{K-1}{w}\rfloor}\sum_{k=(\mathcal{E}-1)w}^{\mathcal{E}w} B_r^{(k-w:k)}(\Delta_\pi) \le wB_r(\Delta_\pi)$, and the inequality (D.73) holds by Lemma 9. □

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

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

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 every $(s,a) \in \mathcal{S}\times\mathcal{A}$, $\Lambda_w^{(k)}(s,a) = \varphi(s,a)(\bar{\Lambda}_w^k)^{-1}\varphi(s,a)^\top$ holds. Now, the following term can be bounded as

$$\begin{aligned}
&\sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\sqrt{\Lambda_w^{(k)}(s_h^{(k+1)}, a_h^{(k+1)})} \\
&= \sum_{k=1}^{K-1}\sum_{h=0}^{H-1}\sqrt{\varphi(s_h^{(k+1)}, a_h^{(k+1)})(\bar{\Lambda}_w^k)^{-1}\varphi(s_h^{(k+1)}, a_h^{(k+1)})^\top} \\
&= \sum_{\mathcal{E}=1}^{\lfloor\frac{K-1}{w}\rfloor}\sum_{k=(\mathcal{E}-1)w+1}^{\mathcal{E}w}\sum_{h=0}^{H-1}\sqrt{\varphi(s_h^{(k+1)}, a_h^{(k+1)})(\bar{\Lambda}_w^k)^{-1}\varphi(s_h^{(k+1)}, a_h^{(k+1)})^\top} \\
&\le \sum_{\mathcal{E}=1}^{\lfloor\frac{K-1}{w}\rfloor}\sqrt{Hw}\sqrt{\sum_{k=(\mathcal{E}-1)w+1}^{\mathcal{E}w}\sum_{h=0}^{H-1}\varphi(s_h^{(k+1)}, a_h^{(k+1)})(\bar{\Lambda}_w^k)^{-1}\varphi(s_h^{(k+1)}, a_h^{(k+1)})^\top} \qquad \text{(D.74)} \\
&\le \sum_{\mathcal{E}=1}^{\lfloor\frac{K-1}{w}\rfloor}\sqrt{Hw}\sqrt{\log\left(\frac{\det(\Lambda_w^{\mathcal{E}w+1})}{\det\left((\Lambda_w^{(\mathcal{E}-1)w+2})^{(1)}\right)}\right)} \qquad \text{(D.75)} \\
&\le \left\lfloor\frac{K-1}{w}\right\rfloor\sqrt{Hw}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)} \qquad \text{(D.76)} \\
&\le (K-1)\sqrt{\frac{H}{w}}\sqrt{\log\left(\frac{\lambda+wH}{\lambda}\right)}
\end{aligned}$$

The inequality (D.74) holds by the Cauchy–Schwarz inequality, (D.75) holds by Lemmas (D.1) and (D.2) in [39], and (D.76) holds since $(\Lambda_w^{(\mathcal{E}-1)w+2})^{(1)} \ge \lambda$ and $\Lambda_w^{\mathcal{E}w+1} \le \lambda + wH$. □

Lemma 10 (Upper bound on $\Delta^p_{(k)}(s,a)$)**.** *For every $(s,a) \in \mathcal{S}\times\mathcal{A}$ and given $\delta \in (0,1)$, the following holds with probability at least $1-\delta$:*

$$\Delta^p_{(k)}(s,a) \le B_p^{(k-w+1:k)} + (\Lambda_w^{(k)}(s,a))^{1/2}\cdot|\mathcal{S}|\cdot\sqrt{\frac{H^2}{2}\log\left(\frac{H}{\delta\lambda}\right)} + \lambda\Lambda_w^{(k)}(s,a)$$

Proof of lemma 10. For every $(s,a)\in\mathcal{S}\times\mathcal{A}$, one can write:

$$\begin{aligned}
&\Delta^p_{(k)}(s,a)\\
&= \|P_{(k+1)}(\cdot|s,a) - \widehat{P}_{(k+1)}(\cdot|s,a)\|_1\\
&= \|o^p_{(k+1)}(\cdot,s,a) - \widehat{o}^p_{(k+1)}(\cdot,s,a)\|_1\\
&= \sum_{s'\in\mathcal{S}}\left|\frac{\sum_{t=k-w+1}^k n_t(s',s,a)}{\lambda+\sum_{t=k-w+1}^k n_t(s,a)} - o^p_{(k+1)}(s',s,a)\right|\\
&= \Lambda_w^{(k)}(s,a)\sum_{s'\in\mathcal{S}}\left|\sum_{t=k-w+1}^k n_t(s',s,a) - \left(\lambda+\sum_{t=k-w+1}^k n_t(s,a)\right)o^p_{(k+1)}(s',s,a)\right|\\
&\le \Lambda_w^{(k)}(s,a)\sum_{s'\in\mathcal{S}}\left(\left|\sum_{t=k-w+1}^k\left(n_t(s',s,a) - n_t(s,a)o^p_{(k+1)}(s',s,a)\right)\right| + \left|\lambda o^p_{(k+1)}(s',s,a)\right|\right)\\
&\le \Lambda_w^{(k)}(s,a)\sum_{s'\in\mathcal{S}}\left|\sum_{t=k-w+1}^k\left(n_t(s',s,a) - n_t(s,a)o^p_{(k+1)}(s',s,a)\right)\right| + \lambda\Lambda_w^{(k)}(s,a) \qquad \text{(D.77)}
\end{aligned}$$

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

$$\begin{aligned}
n_t(s',s,a) &= \sum_{h=0}^{H-1}\mathbb{1}\left[(s',s,a) = (s^t_{h+1}, s^t_h, a^t_h)\right]\\
&= \sum_{h=0}^{H-1}\mathbb{1}\left[(s,a) = (s^t_h,a^t_h)\right]\cdot\mathbb{1}\left[s' = s^t_{h+1}\right] \qquad \text{(D.78)}\\
n_t(s,a) &= \sum_{h=0}^{H-1}\mathbb{1}\left[(s,a) = (s^t_h,a^t_h)\right] \qquad \text{(D.79)}
\end{aligned}$$

where $\mathbb{1}[\cdot]$ is an indicator function. Substituting (D.78) and (D.79) into (D.77) yields that

$$\begin{aligned}
&\Lambda_w^{(k)}(s,a)\sum_{s'\in\mathcal{S}}\left|\sum_{t=k-w+1}^k\left(n_t(s',s,a) - n_t(s,a)o^p_{(k+1)}(s',s,a)\right)\right|\\
&= \Lambda_w^{(k)}(s,a)\sum_{s'\in\mathcal{S}}\left|\sum_{t=k-w+1}^k\left(\sum_{h=0}^{H-1}\mathbb{1}\left[(s,a) = (s^t_h,a^t_h)\right]\cdot\mathbb{1}\left[s' = s^t_{h+1}\right]\right.\right.\\
&\left.\left. - \sum_{h=0}^{H-1}\mathbb{1}\left[(s,a) = (s^t_h,a^t_h)\right]\cdot o^p_{(k+1)}(s',s,a)\right)\right|\\
&= \Lambda_w^{(k)}(s,a)\sum_{s'\in\mathcal{S}}\left|\sum_{t=k-w+1}^k\left(\sum_{h=0}^{H-1}\mathbb{1}\left[(s,a) = (s^t_h,a^t_h)\right]\left(\mathbb{1}\left[s' = s^t_{h+1}\right] - o^p_{(k+1)}(s',s,a)\right)\right)\right|\\
&\le \underbrace{\Lambda_w^{(k)}(s,a)\sum_{s'\in\mathbb{S}}\left|\sum_{t=k-w+1}^k\left(\sum_{h=0}^{H-1}\mathbb{1}\left[(s,a) = (s^t_h,a^t_h)\right]\left(\mathbb{1}\left[s' = s^t_{h+1}\right] - o^p_t(s',s,a)\right)\right)\right|}_{\text{(2.1)}}\\
&+ \underbrace{\Lambda_w^{(k)}(s,a)\sum_{s'\in\mathbb{S}}\left|\sum_{t=k-w+1}^k\left(\sum_{h=0}^{H-1}\mathbb{1}\left[(s,a) = (s^t_h,a^t_h)\right]\left(o^p_t(s',s,a) - o^p_{(k+1)}(s',s,a)\right)\right)\right|}_{\text{(2.2)}}
\end{aligned}$$

The term (2.1) can be upperbounded by utilizing the Lemmas (34) and (43) in [31]. For every $t \in [K]$ and $s' \in \mathcal{S}$, we define the random variable $\eta^t(s') \coloneqq \sum_{h=0}^{H-1} \left(\mathbb{1}\left[s' = s_{h+1}^t\right] - o_t^p(s', s_h^t, a_h^t) \right)$. Given $s' \in \mathcal{S}$, the sequence $\{\eta^\tau(s')\}_{\tau=1}^{\infty}$ is a zero-mean and $H/2$-sub Gaussian random variable. From the Lemma 43 in [31], we set $Y = \lambda\mathbb{I}$ and $X_t = \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right]$. Then, for a given $\delta \in (0,1)$, the following holds with probability at least $1-\delta$ for all $(s,a) \in \mathcal{S} \times \mathcal{A}$:

$$
\begin{aligned}
&\left| (\Lambda_w^{(k)}(s,a))^{1/2} \sum_{t=k-w+1}^{k} \left(\sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \cdot \sum_{h=0}^{H-1} \mathbb{1}[s' = s_{h+1}^t] - o_t^p(s', s, a) \right) \right| \\
&\leq \sqrt{\frac{H^2}{2} \log\left(\frac{(\Lambda_w^{(k)}(s,a))^{-1/2} \cdot \lambda^{-1/2}}{\delta/H} \right)} \\
&= \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta} \cdot \frac{1}{(\Lambda_w^{(k)}(s,a))^{1/2} \cdot \lambda^{1/2}} \right)} \\
&\leq \sqrt{\frac{H^2}{2} \log\left(\frac{H}{\delta} \cdot \frac{1}{\lambda} \right)}
\end{aligned} \tag{D.80}
$$

As a result, the following inequality holds with probability at least $1-\delta$:

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

The term (2.2) can be bounded as

$$
\begin{aligned}
(2.2) &\leq \Lambda_w^{(k)}(s,a) \sum_{s' \in \mathcal{S}} \sum_{t=k-w+1}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \left| o_t^p(s', s, a) - o_{(k+1)}^p(s', s, a) \right| \\
&= \Lambda_w^{(k)}(s,a) \sum_{t=k-w+1}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \sum_{s' \in \mathcal{S}} \left| o_t^p(s', s, a) - o_{(k+1)}^p(s', s, a) \right| \\
&= \Lambda_w^{(k)}(s,a) \sum_{t=k-w+1}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \left\| o_t^p(\cdot, s, a) - o_{(k+1)}^p(\cdot, s, a) \right\|_1 \\
&\leq \max_{t \in [k-w+1, k]} \left(\left\| o_t^p(\cdot, s, a) - o_{(k+1)}^p(\cdot, s, a) \right\|_1 \right) \cdot \left(\Lambda_w^{(k)}(s,a) \sum_{t=k-w+1}^{k} \sum_{h=0}^{H-1} \mathbb{1}\left[(s,a) = (s_h^t, a_h^t)\right] \right) \\
&\leq \max_{t \in [k-w+1, k]} \left(\left\| o_t^p(\cdot, s, a) - o_{(k+1)}^p(\cdot, s, a) \right\|_1 \right) \cdot 1 \\
&\leq B_p^{(k-w+1:k)}(\Delta_\pi)
\end{aligned} \tag{D.81}
$$

Then, by combining (D.77), (D.80) and (D.81), the term $\Delta_{(k)}^p(s,a)$ can be expressed as

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

□

Lemma 11 (Upper bound on $\bar{\Delta}_K^p$)**.** *Given $\delta \in (0,1)$, the following inequality holds with probability at least $1-\delta$:*

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

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

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

□

Proof of Theorem 4 . Before introduing the proof, we first go over some details about Theorem 4 in the following paragraph.

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

$$
\min_{\phi_f^\diamond, q} \mathcal{L}\left(\phi_f^\diamond, q\,;\, \square_{1:N}\right) \text{ where } \mathcal{L}\left(\phi_f^\diamond, q\,;\, \square_{1:N}\right) = \sum_{t=1}^{N} q_t \left(\widehat{\square}_{\phi_f^\diamond}^{k+1} - \square_t\right)^2 + \text{disc}(q) + \frac{1}{wH}\cdot\lambda\|\phi_f^\diamond\|_2 \tag{D.82}
$$

where $\diamond = r$ or p. If $\diamond = r$, then $\square = R(s,a)$ and if $\diamond = p$, then $\square = P(s',s,a)$. Moreover, $\square_{\phi_f^\diamond}$ 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 upper-bounded by the 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 $\{R_1(s,a), R_1(s,a), \ldots, R_N(s,a)\}$ that the agent has received when visiting (s,a). We demonstrate a modified upper bound on $\mathfrak{R}_I$ when utilizing `W-LSE`. To do so, we define the forecasting reward model error $\Delta_{r,k}^1(s,a) = \left|\left(R_{(k+1)} - \widetilde{R}_{(k+1)}\right)(s,a)\right|$ and the forecasting transition probability model error as $\Delta_{(k)}^p(s,a) = \left\|\left(P_{(k+1)} - \widehat{P}_{(k+1)}\right)(\cdot \mid s,a)\right\|_1$ where $\widetilde{R}_{(k+1)}$ and $\widehat{P}_{(k+1)}$ are predicted reward,transition probability from function $g \circ f$ (Appendix D.2).

We now brought the Theorem 7 of [22] to offer an upper bound on the l_2-norm of the reward gap between $R_{(k+1)}(s,a)$ and $\widetilde{R}_{(k+1)}(s,a)$ as follows. To this end, we 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 that the environment provides the agent with a noisy reward $\widehat{Y}_{k,h} = Y_{k,h} + \eta$, where η is sampled from a zero-mean Gaussian. 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 in Section D.1.1. Now, we set $r(x) = c^\top \varphi(x)$ where the vector $c \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ is the same as the estimated future reward vector $\widetilde{R}_{(k+1)} \in$

$\mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ and $r(x)$ is the same as the estimated future reward when $x = (s,a)$, namely $\widetilde{R}_{(k+1)}(s,a)$. Then, for data until episode k, i.e., $\mathcal{D}_{data} = \{(X_{1,0}, \widehat{Y}_{1,0}), (X_{1,1}, \widehat{Y}_{1,1}), .., (X_{k,H-1}, \widehat{Y}_{k,H-1})\}$, we denote $\mathcal{D}_{data}^{(s,a)} \coloneqq \{(X_{k,h}, \widehat{Y}_{k,h}) \mid X_{k,h} = (s,a) \text{ such that } (X_{k,h}, \widehat{Y}_{k,h}) \in \mathcal{D}_{data}\}$. We relabel $\mathcal{D}_{data}^{(s,a)}$ as $\{\big((s,a), \widehat{Y}_1\big), \big((s,a), \widehat{Y}_2\big), ..., \big((s,a), \widehat{Y}_N\big)\}$ such that $N(s,a) = \sum_{t=1}^{k} n_t(s,a)$ is the total number of visitations of (s,a) until episode k (Definition (D.79)). We use the shorthand notation N as $N(s,a)$, and $\sum_{t=1}^{N} q_t = 1$. For every $(s,a) \in \mathcal{S} \times \mathcal{A}$, the following inequalities hold with probability at least $1-\delta$ for all functions $r \in \{x \to c^\top \Psi(x) : \|c\|_2 \le \Lambda\}$:

$$\mathbb{E}[\big(r(s,a) - \widehat{Y}_{N+1}\big)^2 | \mathcal{D}_{data}^{(s,a)}] \le \sum_{t=1}^{N} q_t \big(r(s,a) - \widehat{Y}_t\big)^2 + \mathrm{disc}(q) + \frac{1}{wH} \cdot \lambda \|\bar{r}\|_2 \tag{D.83}$$

Take the expectation over η on both inequailty.

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

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

The left-hand side of (D.83) can be expressed as

$$\begin{aligned}\mathbb{E}\big[(r(s,a) - \widehat{Y}_{N+1} - \eta)^2\big] &= \mathbb{E}_\eta\big[(r(s,a) - Y_{N+1})^2\big] + \mathbb{E}_\eta\big[\eta^2\big] \\ &= (r(s,a) - Y_{N+1})^2 + \mathbb{E}\big[\eta^2\big]\end{aligned} \tag{D.84}$$

Also, the term $\sum_{t=1}^{N} \mathbb{E}_\eta \Big[q_t \big(r(s,a) - \widehat{Y}_t\big)^2\Big]$ of the right-hand side of equation (D.83) can be written as

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

By eliminating $\mathbb{E}_\eta[\eta^2]$ from both sides, we obtain that

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

Recall the definition of $r(s,a) = \widetilde{R}_{(k+1)}(s,a)$, $Y_t = R_t(s,a)$. Since t matches one of $(k,h) \in [K] \times [H]$ pairs, we can rewrite

$$\begin{aligned}\sum_{t=1}^{N} q_t \big(r(s,a) - \widehat{Y}_t\big)^2 &= \sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \big(r(s,a) - Y_{(k,h)}\big)^2 \\ &= \sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \Big(\widetilde{R}_{(k+1)}(s,a) - R_h^{k'}(s,a)\Big)^2 \\ &= \sum_{k'=1}^{k-1} \sum_{h=0}^{H-1} q_{(k',h)} \Big(\widetilde{R}_{(k+1)}(s,a) - R^{k'}(s,a)\Big)^2\end{aligned}$$

where if (s,a) is not visited at step h of episode k, then the corresponding $q_{(k',h)}$ is zero. As a result,

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

A similar analysis for $\Delta^p_{(k)}$ leads to the following inequality for all $(s', s, a) \in \mathcal{S} \times \mathcal{S} \times \mathcal{A}$:

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

On the other hand,

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

Recall the Corollary 5, Corollary6 and $\mathfrak{R}_I$ definition. Aftering fixing (s, a), the term $\mathfrak{R}_I(s, a)$ can be expressed as

$$\begin{aligned}
\mathfrak{R}_I =& \frac{1}{1-\gamma} \sum_{k=1}^{K-1} \iota_\infty^{k+1} + \sum_{k=1}^{K-1} \sum_{h=0}^{H-1} -\iota_h^{(k+1)} + C_p \sqrt{K-1} \\
\leq& \frac{1}{1-\gamma} \sum_{k=1}^{K-1} \left(\Delta^r_{(k)}(s,a) + \Delta^p_{(k)}(s,a) \frac{\gamma \hat{r}_{max}}{1-\gamma} - 2\Gamma^{(k)}_w(s,a) \right) \\
&+ \sum_{k=1}^{K-1} \sum_{h=0}^{H-1} \left(\Delta^r_{(k)}(s,a) + \Delta^p_{(k)}(s,a) \frac{\gamma \hat{r}_{max}}{1-\gamma} + 2\Gamma^{(k)}_w(s,a) \right) \\
&+ C_p \sqrt{K-1} \\
\leq& \frac{1}{1-\gamma} \sum_{k=1}^{K-1} \left(\Delta^r_{(k)}(s,a) + \Delta^p_{(k)}(s,a) \frac{\gamma}{1-\gamma} \left(\tilde{r}_{\max} + \max(2\Gamma^{(k)}_w(s,a)) \right) - 2\Gamma^{(k)}_w(s,a) \right) \\
&+ H \sum_{k=1}^{K-1} \left(\Delta^r_{(k)}(s,a) + \Delta^p_{(k)}(s,a) \frac{\gamma}{1-\gamma} \left(\tilde{r}_{\max} + \max(2\Gamma^{(k)}_w(s,a)) \right) + 2\Gamma^{(k)}_w(s,a) \right) \\
&+ C_p \sqrt{K-1} \\
\leq& \sum_{k=1}^{K-1} \Bigg(\underbrace{\left(\frac{1}{1-\gamma} + H \right) \Delta^r_{(k)}(s,a) + \frac{\gamma \tilde{r}_{\max}}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \Delta^p_{(k)}(s,a)}_{①} + \\
&+ \frac{\gamma}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \max(2\Gamma^{(k)}_w(s,a)) \Delta^p_{(k)}(s,a) \Bigg) \\
&+ \sum_{k=1}^{K-1} 2 \left(-\frac{1}{1-\gamma} + H \right) \Gamma^{(k)}_w(s,a) \\
&+ C_p \sqrt{K-1}
\end{aligned}$$

We set the term ① to be $2(\frac{1}{1-\gamma} + H)\Gamma^{(k)}_w(s,a)$, which requires redefining the exploration bonus term as

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

Also, note that $\Delta^p_{(k)}(s,a) = \sum_{s' \in \mathcal{S}} \left| \left(\widehat{P}_{(k+1)} - P_{(k+1)} \right)(s'|s,a) \right| \leq |\mathcal{S}|$. Therefore,

$$
\begin{aligned}
\mathfrak{R}_I &\leq \sum_{k=1}^{K-1} \left(4H\Gamma^{(k)}_w(s,a) + \frac{2\gamma}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \max(\Gamma^{(k)}_w(s,a)) \, |\mathcal{S}| \right) \\
&\leq \sum_{k=1}^{K-1} \left(4H + \frac{2\gamma}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \right) \max(\Gamma^{(k)}_w(s,a)) \\
&= \left(4H + \frac{2\gamma \, |\mathcal{S}|}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \right) \sum_{k=1}^{K-1} \max(\Gamma^{(k)}_w(s,a)) \\
&\leq \left(4H + \frac{2\gamma \, |\mathcal{S}|}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \right) \sum_{k=1}^{K-1} \left(\frac{1}{2} \max(\Delta^r_{(k)}(s,a)) + \frac{\gamma \tilde{r}_{\max}}{2(1-\gamma)} \max(\Delta^p_{(k)}(s,a)) \right) \\
&= \left(4H + \frac{2\gamma \, |\mathcal{S}|}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \right) \sum_{k=1}^{K-1} \left(\frac{1}{2} \Delta^r_{(k)}(s,a) + \frac{\gamma \tilde{r}_{\max}}{2(1-\gamma)} \Delta^p_{(k)}(s,a) \right) \\
&= \left(4H + \frac{2\gamma \, |\mathcal{S}|}{1-\gamma} \left(\frac{1}{1-\gamma} + H \right) \right) \left(\frac{1}{2} \sum_{k=1}^{K-1} \Delta^r_{(k)}(s,a) + \frac{\gamma \tilde{r}_{\max}}{2(1-\gamma)} \sum_{k=1}^{K-1} \Delta^p_{(k)}(s,a) \right)
\end{aligned}
$$

Note that above upper bound on $\mathfrak{R}_I$ holds under the following conditions for $\Delta^r_{(k)}(s,a)$ and $\Delta^p_{(k)}(s,a)$:

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

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

□

Proof of Remark 1. The proof starts with (D.85). Define

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

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

where r_{sw} is the same reward estimation as in (D.7). Then the minimum of (D.83) yields that

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

The term ① is the optimization problem in (D.86) whose minimizer is r^{sw}. An inspection of (D.87) and (D.88) concludes that the optimal solution $(q^*, \bar{r}^*)$, namely the minimizer of (D.87) provides a smaller value than (q^{sw}, r^{sw}). Since the right-hand side (D.85) is same as (D.87), $(q^*, \bar{r}^*)$ provides a tighter upper bound on the left-hand side term of equation (D.83) than q^{sw}, r^{sw}. Therefore, (D.84) implies that the optimal solution $(q^*, \bar{r}^*)$ gives a tighter upper bound on $\Delta^r_{(k)}$ than using $(q^{sw}, \bar{r}^{sw})$. One can repeat the above argument for the upper bound on $\Delta^p_{(k)}$. Then, by Corollary 5 and 6, the tighter upper bounds on $\Delta^r_{(k)}(s,a)$ and $\Delta^p_{(k)}(s,a)$ provide smaller upper bounds on $-\iota^{(k)}_H, \bar{\iota}^K_\infty$ and lead to a tighter upper bound on $\mathfrak{R}_I$. □

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 $R = R_h + R_f - R_c$, where R_h is the healthy reward, $R_f = k_f(x_{t+1} - x_t)/\Delta t, k_f > 0$ is the forward reward, and R_c is the control cost. The agents have a goal to run faster in the $+x$ direction, and therefore the faster they run, the higher the forward reward R_f is. We modify the environment to make the agent's desired directions change as the episode goes by. To be specific, we design the forward reward R_f to change as episodes progress in the form of $R_f^k = o_k \cdot k_f(x_{t+1} - x_t)/\Delta t$ where $o_k = a\sin(wbk)$ and k is a episode where $a, b > 0$ are constants. A positive o_k causes the agent to desire a forward $+x$ direction as an optimal policy, and a negative o_k causes it to desire a backward $-x$ direction. We generate different speeds of non-stationarity by changing the frequency variable $w \in \{1, 2, 3, 4, 5\}$.

Non-stationary variable o_k generator.

1. Sine function: The non-stationary parameter 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 number. We change w in the set $[1, 2, 3, 4, 5]$. We divide $2\pi wk$ by 37, a prime number, to ensure that the environment has various non-stationary modes and to avoid certain non-stationary parameters appearing frequently.
2. Real data: we bring the stock price data to model a non-stationary real dataset.

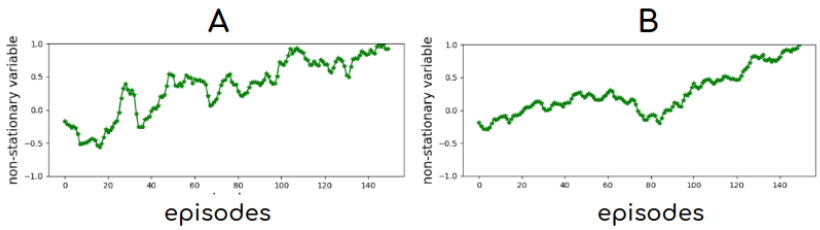

Figure 4: Nonstationary parameter from real data A,B

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

	$G = 38$	$G = 76$	$G = 114$	$G = 152$	$G = 190$
$B(G)$	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 use the auto_arima function of pmdarima python package to find the optimal p, q, d. To compare the results between `ProST-G` and MBPO, we train the MBPO and `ProST-G` with the initial learning rate $lr = 0.0003$ with the decaying parameter 0.999. For `ProST-G`, We add the uniform noise $\eta \sim \text{Unif}([-b, b])$ to the non-stationary parameter o^k to generate the noisy non-stationary parameter $\hat{o}_k = o_k + \eta$ with different noise bounds $b \in \{0.01, 0.03, 0.05\}$. We denote $\text{Unif}([-b, b])$ as continuous uniform distributions over the interval $[-b, b]$.

To compare the results between `ProST-G` and ProOLS, ONPG, FTML, we train these three baselines with eight different initial learning rates $lr \in \{0.001, 0.003, 0.005, 0.007, 0.01, 0.03, 0.05, 0.07\}$.

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: sine wave

(1) Swimmer-v2

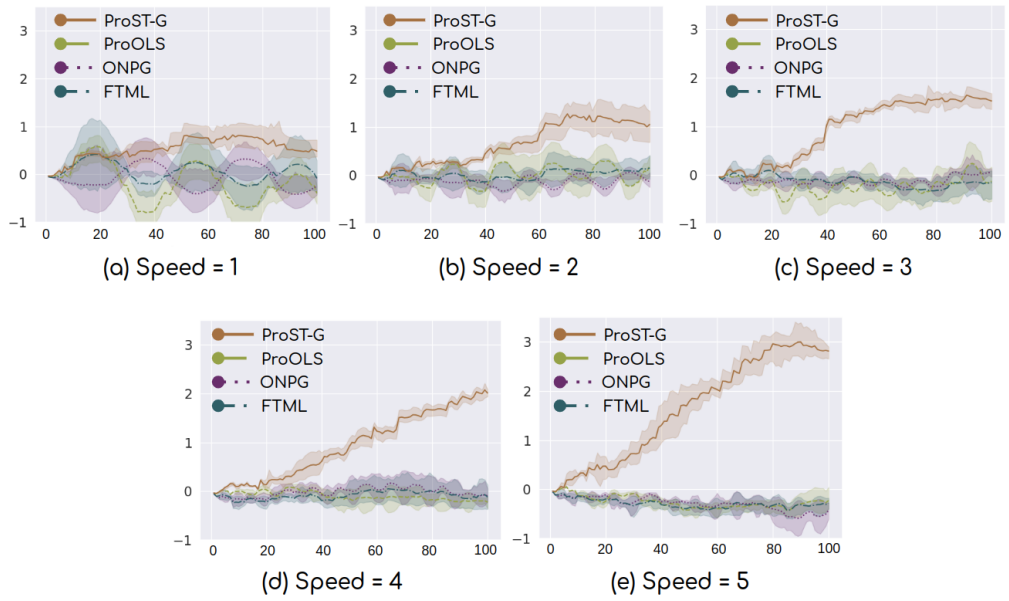

Figure 5: (a) ~ (e) the average rewards of ProST-G, and the three baselines: ProOLS, ONPG, FTML for 5 different speeds (x-axis indicates the episode). The shaded area of ProST-G is 95% confidence area among 3 different noise bounds, and the shaded areas of three baselines are the 95 % confidence area among 8 different learning rates.

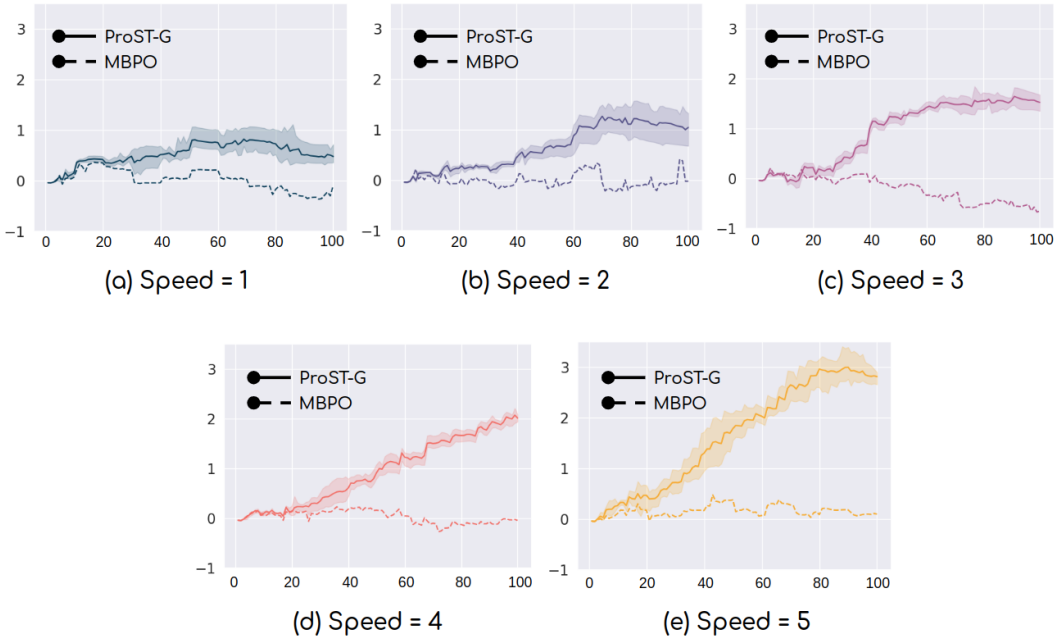

Figure 6: (a) ~ (e) the average rewards of ProST-G and MBPO. The shaded area of ProST-G is 95% confidence area among 3 different noise bounds.

(2) Halfcheetah-v2

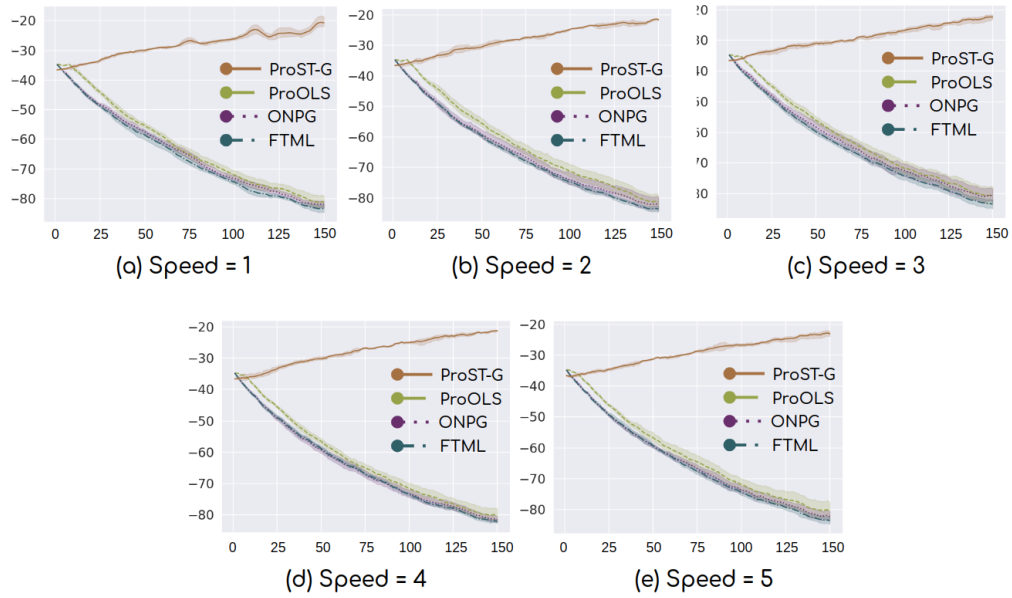

Figure 7: (a) ~ (e) the average rewards of ProST-G, and the three baselines: ProOLS, ONPG, FTML for 5 different speeds (x-axis indicates the episode). The shaded area of ProST-G is 95% confidence area among 3 different noise bounds, and the shaded ares of three baselines are the 95% confidence areas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

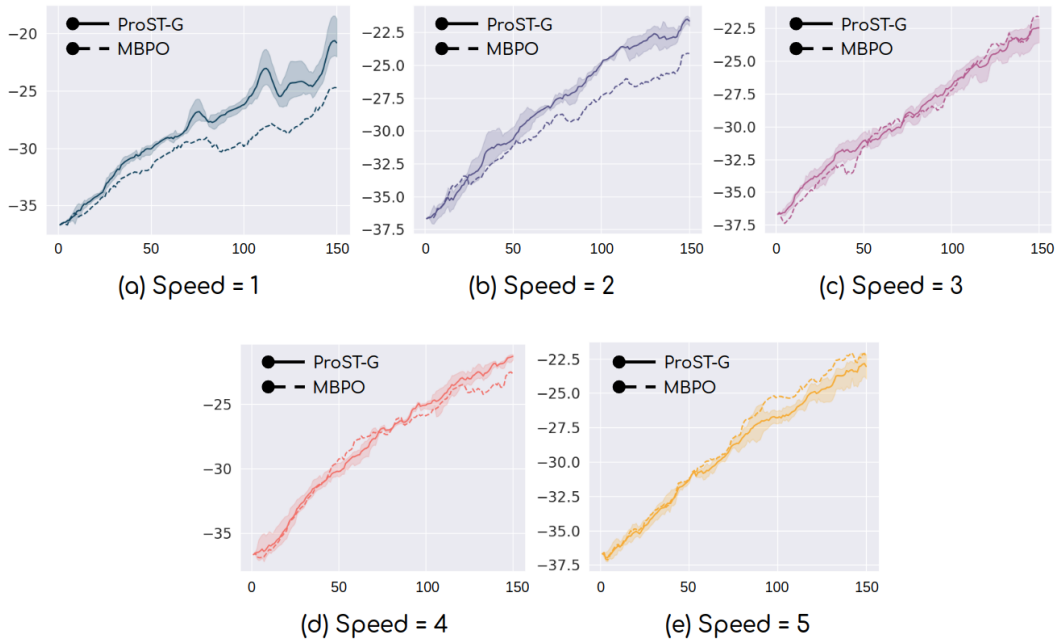

Figure 8: (a) ~ (e) the average rewards of ProST-G and MBPO (x-axis indicates the episode). The shaded area of ProST-G is 95% confidence area among 3 different noise bounds.

(3) Hopper-v2

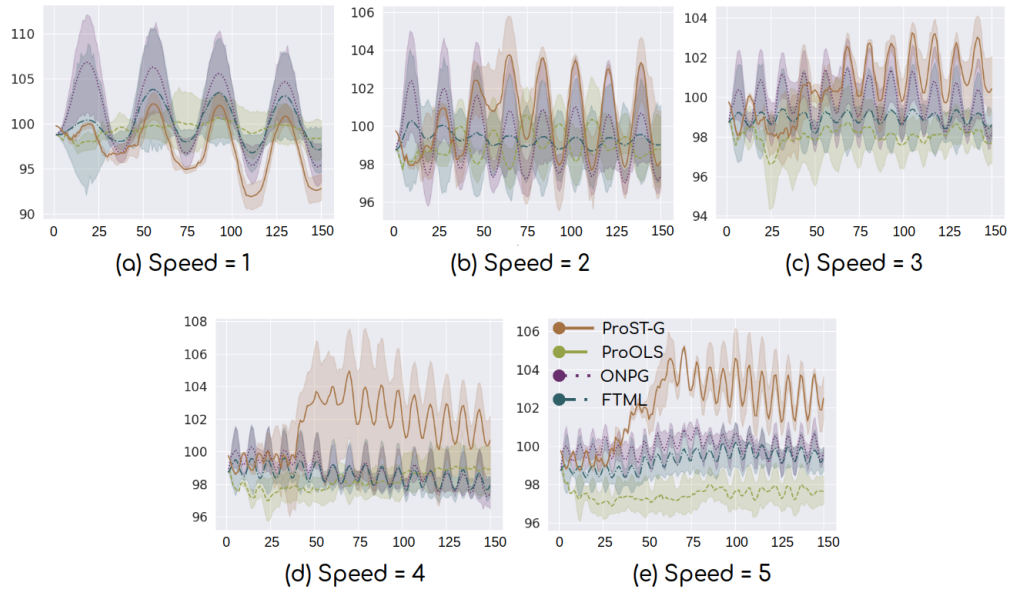

Figure 9: (a) ~ (e) the average rewards of `ProST-G`, and the three baselines : ProOLS, ONPG, FTML for 5 different speeds (x-axis indicates the episode). The shaded area of `ProST-G` is 95% confidence area among 3 different noise bounds, and the shaded areas of three baselines are the 95% confidence areas among 8 different learning rates.

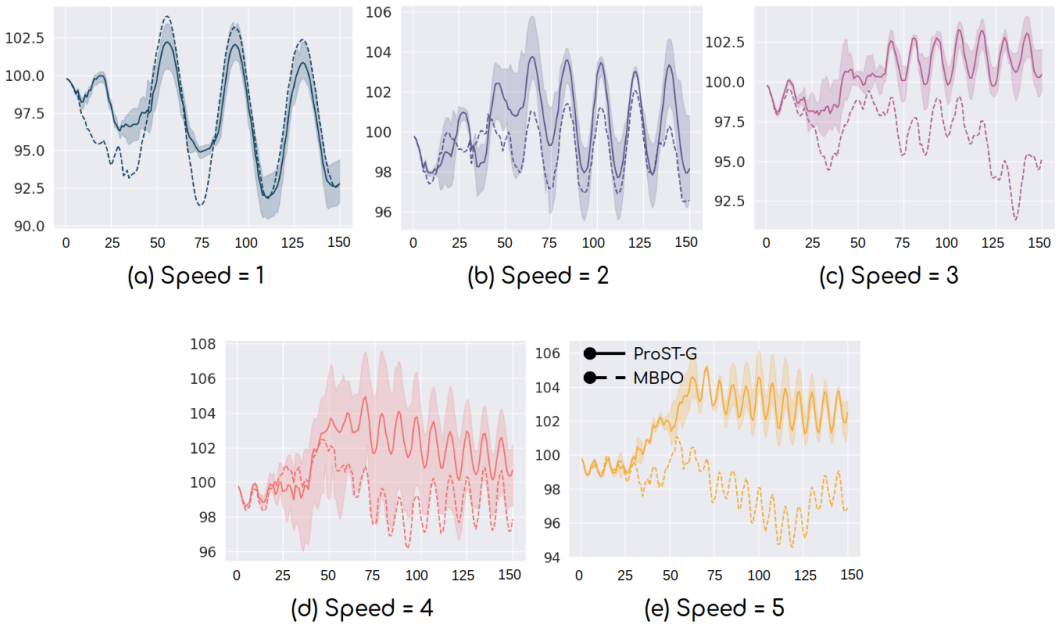

Figure 10: (a) ~ (e) the average rewards of `ProST-G` and MBPO (x-axis indicates the episode). The shaded area of `ProST-G` is 95% confidence area among 3 different noise bounds.

E.3.2 Non-stationarity : real data

The shaded area of ProST-G is 95% confidence area among 3 different noise bounds, and the shaded ares of three baselines are the 95% confidence area among 8 different learning rates.

(1) Swimmer-v2

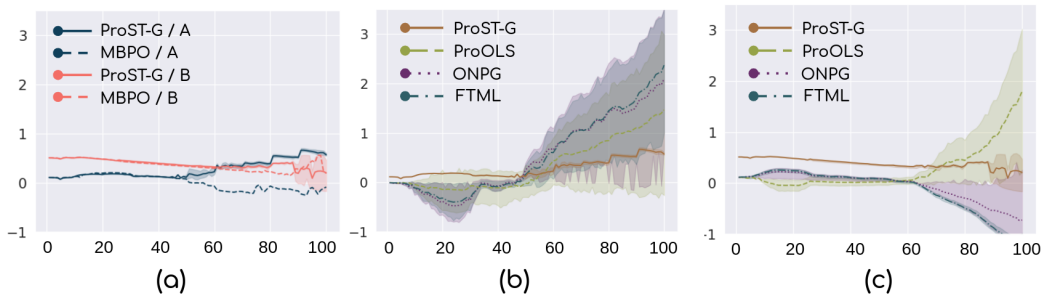

Figure 11: (a) average reward with ProST-G and MBPO on real data A,B (x-axis is episode). (b) average reward with ProST-G and three baselines on realdata A. (c) average reward with ProST-G and three baselines on realdata B.

(2) Halfcheetah-v2

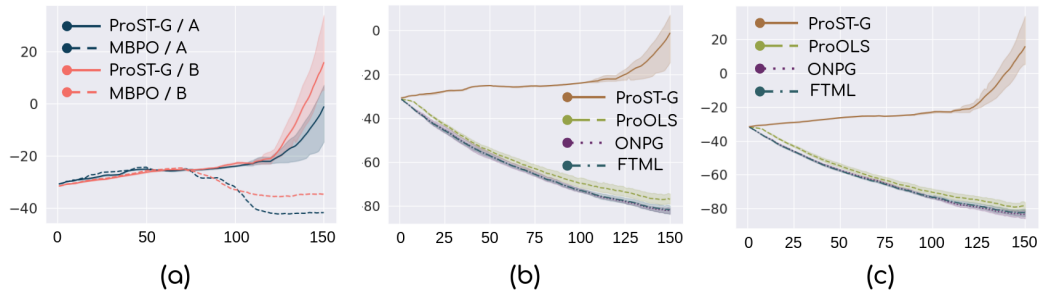

Figure 12: (a) average reward with ProST-G and MBPO on real data A,B (x-axis is episode). (b) average reward with ProST-G and three baselines on realdata A. (c) average reward with ProST-G and three baselines on realdata B.

(3) Hopper-v2

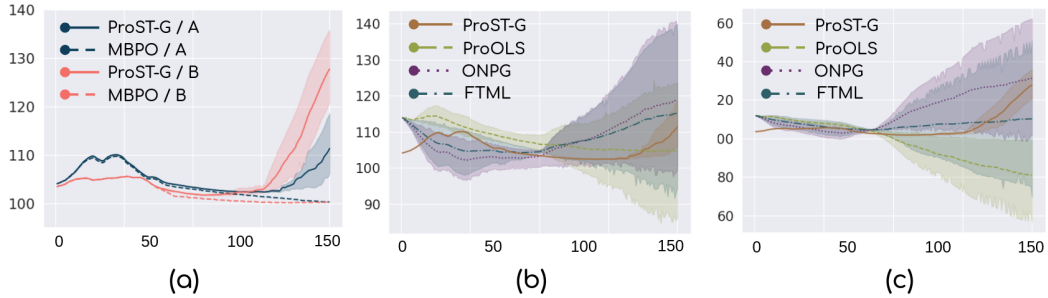

Figure 13: (a) average reward with ProST-G and MBPO on real data A,B (x-axis is episode). (b) average reward with ProST-G and three baselines on realdata A. (c) average reward with ProST-G and three baselines on realdata B.

F Algorithms

F.1 ProST framework

Algorithm 1: ProST framework

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 $\mathcal{D}_{env}$.
  /* MDP forecaster $g \circ f$ */
  /* (1) Observe and forecast: */
  Observe a noisy non-stationary parameter $\hat{o}_k$
  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 $\widehat{\mathcal{M}}_{k+1}$ 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: 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 parameter $\hat{o}_k$
  /* MDP forecaster $g \circ f$: (1) update f, g */
  Update $f_{\phi_f} : \phi_f^k \leftarrow \arg\min_\phi \mathcal{L}_f(\hat{o}_{k-(w-1):k}; \phi)$
  Update $g_k^P(s', s, a, o)$
  Update $g_k^R(s, a, o)$
  /* MDP forecaster $g \circ f$: (2) predict $\widehat{\mathcal{P}}^{k+1}, \widehat{\mathcal{R}}^{k+1}$ */
  Forecast 1 episode ahead non-stationary parameter: $\hat{o}_{k+1} = f_{\phi_f^k}(\hat{o}_{k-(w-1):k})$
  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 regularization */
  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 observation $\widehat{o}_{k-(w-1):k}$. The ARIMA model is one of the most general classes 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 $X_t - \alpha_1 X_{t-1} - \cdots - \alpha_p X_{t-p} = \epsilon_t + \theta_1 \epsilon_{t-1} + \cdots + \theta_q \epsilon_{t-q}$, where α_i's are the parameters of the autoregressive part of the model, the θ_i's are the parameters of the moving average part, and ϵ_t's are the error terms that take d times difference between X_ts, which we assume to be independent and follow a normal distribution with a zero mean.

(2) Model predictor g. We use a bootstrap ensemble of dynamic models $\{g^1_{\phi_g}, g^2_{\phi_g}, ..., g^M_{\phi_g}\}$. 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, namely $g^i_{\phi_g}(s_{h+1}, r_h | s_h, a_h, \widehat{o}_{k+1}) = \mathcal{N}(\mu^i_{\phi_g}(s_h, a_h), \Sigma^i_{\phi_g}(s_h, a_h))$. To efficiently handle uncertainty due to the non-stationary environment, we design each neural network to be a probabilistic model to capture the aleatoric uncertainty, i.e. the noise of the output, and learn multiple models as bootstrap ensemble to handle the epistemic uncertainty, i.e. the uncertainty in the model parameters. Then we predict s_{h+1} and 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 the 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: 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 $\mathcal{D}_{env}$.
  /* MDP forecaster $g \circ f$: (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 $\mathcal{D}_{env}$
  /* MDP forecaster $g \circ f$: (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`.