Reinforcement Learning with Multi-Step Lookahead Information Via Adaptive Batching

Nadav Merlis
Technion – Israel Institute of Technology
nmerlis@technion.ac.il

Abstract

We study tabular finite-horizon reinforcement learning problems with multiple steps of lookahead information. Before acting, the learner observes ℓ steps of future transition and reward realizations: the exact state the agent would reach and the rewards it would collect under any possible course of action. While it has been shown that such information can drastically boost the value, finding the optimal policy is NP-hard, and it is common to apply one of two tractable heuristics: processing the lookahead in chunks of predefined sizes ('fixed batching policies'), and model predictive control. We first illustrate the problems with these two approaches and propose utilizing the lookahead in adaptive (state-dependent) batches; we refer to such policies as adaptive batching policies (ABPs). We derive the optimal Bellman equations for these strategies and design an optimistic regret-minimizing algorithm that enables learning the optimal ABP when interacting with unknown environments. Our regret bounds are order-optimal up to a potential factor of the lookahead horizon ℓ, which can usually be considered a small constant.

1 Introduction

In reinforcement learning (RL), an agent sequentially interacts with potentially unknown environments, gaining rewards or penalties for its interaction [24]. The instantaneous rewards are influenced by the agent's action, as well as by the state of the environment, which in turn changes due to the agent's behavior. As such, to maximize the cumulative rewards, agents should not only maximize their immediate gain, but also take into consideration how their actions affect the environment. In the standard model, the agent relies on the current state of the environment to pick an action, and then observes its outcome: the reward it earns and the new state of the environment.

We study problems in which agents obtain additional valuable information before acting, in the form of *lookahead* information. That is, agents get to observe the *realized outcomes* for all their potential actions ℓ steps into the future: both where they would transition to and what rewards they would collect. For example, in navigation, traffic information predicts with high certainty which roads are available and how long it would take to traverse them; in packing problems, the agent often observes the next few items to be packed; in resource management, nearby supply and demand can be accurately estimated; and so on. This type of information has been shown to greatly boost the potential reward that the agent may collect: assuming A actions and interaction length H, the value can increase by a factor of at most $\approx A^{\ell}$ for reward information [17], and of $\approx A^{\Omega(H)}$ with even one step of transition information [16]. However, correctly utilizing this information is tricky due to its exponential nature. Specifically, while one can efficiently plan and learn with 1-step lookahead information [16], it has been proven in discounted settings that exact optimal planning with 2-step lookahead is NP-hard [19], and the best known approximation algorithms are exponential in ℓ [20].

In this work, we propose a family of tractable policies for tabular finite-horizon RL problems with ℓ-step lookahead information that we term adaptive batching policies (ABPs). Specifically, at each

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

step, these policies decide how many steps to look into the future, depending also on the current state of the environment, and use this information to its end before asking for new information. Such policies clearly improve agents that process information in predefined batches – which we show can catastrophically fail even in extremely simple environments – but are still tractable to optimize via dynamic programming; this is in contrast to model predictive controllers, for which we prove that dynamic programming schemes can be exponentially suboptimal. Specifically, we derive the optimal Bellman equations for ABPs and explain how to efficiently find the optimal policy. We then show that the optimal ABP can also be learned while interacting with an unknown environment in episodes and prove near-tight regret bounds (up to factors of ℓ, which is usually a small constant).

1.1 Related Work

We first emphasize the important distinction between obtaining lookahead information and performing lookahead planning. In control, lookahead often refers to obtaining additional knowledge about the future behavior of the system in the near future: the exact/approximate dynamics, the noise realization in the system, and more. This is the type of lookahead we study in this paper, and we refer to it as *lookahead information*. This information can also be viewed as a *prediction*. For example, in linear control with quadratic costs, and when lookahead reveals zero-mean additive i.i.d. noise, it is well known that model predictive control [MPC, 3] strategies are optimal [e.g., 26]. In particular, previous works have studied the optimality of MPC-like strategies in the presence of lookahead information in various settings, including linear and nonlinear control systems, with stochastic or adversarial state perturbations, inexact information, and more [26, 12, 28, 13, 14].

In contrast, in the context of RL and/or Markov decision processes (MDPs), the term lookahead usually refers to propagating the model for multiple steps and/or sampling trajectories of a certain length as part of a planning or value-learning procedure [25, 8, 18, 22, 4]. Specifically, this procedure does not incorporate any new information about the future of the system in the current interaction, but aims to provide more efficient means to calculate the optimal Markov policy. When working with lookahead information, it may be useful to adapt such approaches and perform multi-step planning as a subroutine, but this line of work studies settings that are fundamentally different from ours.

In tabular RL with lookahead information, [17] analyzed how the optimal value increases due to multiple steps of future reward information, as a function of the lookahead range and properties of the environment. However, they did not propose any algorithm for interaction. Merlis [16] then studied problems in which agents observe either the immediate rewards or transitions before acting (one-step reward/transition lookahead). They derived the optimal Bellman equations and proposed learning algorithms with order-optimal regret bounds. We show even better rates with combined one-step reward and transition lookahead information and extend both the planning and learning beyond one-step lookahead.

For multi-step lookahead, Pla et al. [19] prove that even with two steps of transition lookahead, exact planning is NP-hard in discounted/average tabular MDPs (and also provide a polynomial-time algorithm for one-step transition lookahead). Moreover, the best known results for approximate planning with multi-step lookahead are exponential in the number of lookahead steps [20]. Consequently, we cannot aim to find the optimal lookahead policy, but rather limit ourselves to adaptive batching policies, which can be efficiently calculated. In [31], the authors study the performance of MPC strategies in nonstationary environments where the agent obtains noisy multi-step predictions on the expected rewards and transition kernels. They show that given a sufficient lookahead horizon and mixing, MPC will converge to the optimal agent. We study a different setting (stochastic environments where we observe future realizations) in different regimes (no mixing assumptions), so the results are incomparable. In particular, we demonstrate how MPC can critically fail in our setting. In addition, [15] studies an offline learning problem in which, given access to a generative model, the goal is to estimate the optimal policy with multi-step noisy transition lookahead information. While there are similarities in the underlying model, we study orthogonal aspects of the problem. For instance, they allow noisy transition lookahead observations, while we study perfect predictions, but also include reward lookahead. Moreover, they assume that information arrives at predefined timesteps $1, \ell+1, 2\ell+1, \ldots$; we explain why this can cause extremely suboptimal behavior and instead let the agent adaptively decide its lookahead range. Finally, both the interaction protocol and objectives are very different. We focus on learning during the interaction and regret minimization, balancing exploration and exploitation. In contrast, [15] assumes access to samples of an equal (predefined)

size from all state-action pairs and lookahead observations, and attempts to output a near-optimal policy using this data.

We end this section by mentioning a small selection from the numerous works on regret analysis in tabular MDPs [10, 1, 11, 5, 27, 7, 9, 23, 29, 30, 16]. Specifically, we use the monotone bonus properties from [29, 30, 16] and some of the techniques in [9, 16].

2 Setting

An episodic tabular Markov decision process (MDP) is defined by the tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{R}, P, H)$, where $\mathcal{S}$ and $\mathcal{A}$ are the state and action spaces (of respective sizes S, A), $\mathcal{R}$ is the reward distribution, P is the transition kernel and H is the interaction horizon. At step h, given that the agent is in state $s_h = s$ and plays an action $a_h = a$, the agent obtains a reward $R_h \sim \mathcal{R}_h(s,a)$, bounded in $[0,1]$, and transitions to the state $s_{h+1} = s'$ with probability $P_h(s'|s,a)$. The rewards and transitions are assumed to be mutually independent between timesteps, but can be arbitrarily correlated between states and actions at a given step h. A policy π prescribes which action the agent takes during the interaction. A deterministic Markov policy $\pi \in \Pi_{det}$ maps each state s and step h to an action $\pi_h(s)$. Policies are measured by their value – their expected cumulative rewards $V_1^\pi(s) = \mathbb{E}\left[\sum_{t=1}^H R_t | s_1 = s, \pi\right]$, where the expectation is w.r.t. both the environment and the policy.

Lookahead information. ℓ-step lookahead information consists of all realized rewards and transition information accessible in ℓ steps of interaction. Formally, given a deterministic policy $\phi \in \Pi_{det}$,[1] we define a trajectory realization starting at $s_h = s$ and ending at t when following $a_h = \phi_h(s_h)$ as

$$\tau_{h:t}^\phi(s) = \{s_h, a_h, R_h, \dots s_{t-1}, a_{t-1}, R_{t-1}, s_t | s_h = s, \phi\}. \tag{1}$$

The ℓ-step lookahead information contains the *realized* trajectories from step h to step $h+\ell$ for all possible policies, that is, $I_{h,\ell}(s) = \left\{\tau_{h:h+\ell}^\phi(s)\right\}_{\phi \in \Pi_{det}}$. We remark that we implicitly assume consistency between realized trajectories: if, in two different realized trajectories, an agent visits a state s_t and plays a_t, then it will necessarily observe the same reward R_t and next state s_{t+1} in both trajectories. We use $\mathcal{I}_{h,\ell}(s)$ to denote the distribution of lookahead information sampled starting at step h and state s, independently from all history up to step h. When clear for the context, we omit the lookahead range ℓ and/or the state s and, e.g., write I_h or $\mathcal{I}_h(s)$. Near the end of the episode, when we have fewer than ℓ remaining steps, we denote the effective lookahead range by $\ell_h = \min\{\ell, H - h + 1\}$.

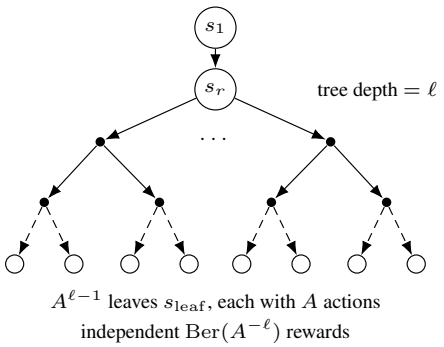
We also denote the reward and state at step t given lookahead information at $s_h = s$ and a policy $\phi \in \Pi_{det}$ as $\mathfrak{R}_{t|h}(s, \phi, I)$ and $\mathfrak{s}_{t|h}(s, \phi, I)$. Specifically for one-step lookahead, the policy is a fixed action, so we can also write $\mathfrak{R}_{h|h}(s, a, I)$ and $\mathfrak{s}_{h+1|h}(s, a, I)$. Finally, for brevity, when the lookahead information I is clear from the context, we sometimes omit it and write, for example, $\mathfrak{R}_{t|h}(s, \phi)$. We describe in depth one potential model for lookahead information generation in Appendix A (though we recommend reading it only after the formal introduction of ABPs in Section 3).

2.1 Lookahead Policies

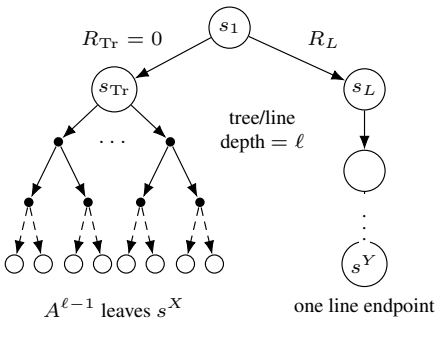
An ℓ-step lookahead policy observes lookahead information at every timestep and decides which action to take based on both the current state s and all future trajectories of length ℓ, that is, $a_h = \pi_h(s_h, I_{h,\ell}(s_h))$. Importantly, I contains an exponential amount of information, and so even with 2 steps of lookahead, finding the optimal policy is NP-hard [19]. One core reason for this hardness is the temporal correlations that lookahead information induces. Specifically, the arrival to state $s_h = s$ may be affected by previous lookahead information that overlaps with $I_{h,\ell}(s)$.

Example 1. *Assume that starting from s_{init}, we can move to either s_1 or s_2. In s_1, there is a Bernoulli reward $R(s_1) \sim \mathrm{Ber}(p)$, but assume that a 2-step lookahead agent π^L decides to go to s_1 only if $R(s_1) = 1$ (and otherwise goes to s_2). Even though for any non-lookahead policy π, it holds that $\mathbb{E}[R(s_1)|s_1, \pi] = p$, when playing the lookahead policy π^L, we have $\mathbb{E}[R(s_1)|s_1, \pi^L] = 1$. Thus, the distribution of the lookahead information from s_1, $\mathcal{I}_{2,2}(s_1)$ is 'distorted' by the policy π^L.*

[1] We emphasize the distinction between a global policy for the entire interaction and a local policy for the duration of the lookahead by denoting the former with π and the latter with ϕ.



(a) Counterexample of Claim 2.1. Fixed batching strategies fail since the state s_1 prevents them from observing the reward at a good timing: for a batching horizon B, rewards will be observed at a distance of $\approx \min\{B, \ell - B\} \leq \ell/2$ from the leaves, which leads to an expected value of $\approx A^{-\ell/2}$.



(b) Counterexample of Claim 2.2. MPC fails since it can be fooled into always heading to the line even though the tree is exponentially better. This is done by creating two candidate reward distributions: Bernoulli and deterministic. Then, we place the distribution with the higher MPC-value at s^Y and the other at s^X.

Figure 1: The diagnostic environments in Claims 2.1 and 2.2. Both environments have deterministic transitions, start at state s_1, and are of horizon $\ell + 1$. Rewards are zero except at the last step (and at the first step in Figure 1b as a tie-breaking mechanism). In both cases, observing Bernoulli rewards $\text{Ber}(A^{-\ell})$ from the root of a full tree exponentially augments their value: to collect a unit reward, only one leaf needs to realize $R = 1$, which happens w.p. $\geq 1 - 1/e$. Observing rewards later in the tree, or not leveraging this exponential increase, is what makes fixed-batching and MPC fail.

Consequently, given s_h, lookahead information is not sampled 'independently' from the distributions $\mathcal{R}, P$, but also has a complex dependence on s_h. One common way to decorrelate the lookahead from the state is to process the information in *batches*. For example, in [15], ℓ-step transition information is observed only at discrete time points $1, \ell + 1, 2\ell + 1, \ldots$, and until each information chunk is exhausted, no new lookahead information is gained. Then, the state in which new lookahead information is gained $s_{n\ell+1}$ only depends on information ending at step $n\ell + 1$, and is not affected in any way by the information in $I_{n\ell+1,\ell}(s_{n\ell+1})$. Therefore, when following such policies, we can assume that the lookahead information is sampled independently from the environment model. More generally, one might decide to process information at predefined intervals of size $B \leq \ell$, so that at time steps $h \in \{nB + 1, \ldots, (n + 1)B\}$, the policy only depends on information at step $nB + 1$: $a_h = \pi_h(s_{nB+1}, I_{nB+1,B}(s_{nB+1}))$. We refer to such policies as *fixed batching policies* and call B the batching horizon or the lookahead range. Unfortunately, there are environments in which all such policies are exponentially suboptimal.

Claim 2.1. *For any $\ell \geq 2, A \geq 2, H \geq \ell + 1$, there exists an environment with $S \leq 1 + A^\ell$ and deterministic transitions s.t. for some fixed initial state, any fixed batching policy collects a value of at most $A^{-\ell/2+1}$ in expectation. In contrast, the optimal ℓ-step lookahead value is larger than $1/2$.*

The proof can be found in Appendix I.1, and a numerical illustration of the values is at Appendix I.3. The environment that causes this behavior is extremely simple: an initial state that deterministically leads to a tree of depth ℓ with Bernoulli rewards at its leaves (see also Figure 1a). An optimal lookahead policy will go to the root of the tree and navigate to one of the few leaves whose reward is realized to be $R = 1$ (if any exist). Yet, we show that for any choice of batching horizon B, we observe the rewards when already being deep inside the tree – we lose most of the navigation power of the agent.

Another prominent candidate designed to handle lookahead information is model predictive control (MPC). The key principle in such controllers is to replace the expected rewards/transition kernel inside the lookahead range by the lookahead realization and plan until the end of the lookahead (with some appropriate value after the lookahead ends). Then, the MPC only plays the first action in the solution, obtains the updated lookahead information (with one additional step), and repeats the entire planning process to incorporate the new information. The most critical issue in these controllers is the value computation; we show that any backwards-induced value calculation necessarily fails

exponentially in some environments; i.e., letting the value only depend on future events can be extremely suboptimal. Specifically, we require the following mild assumption.

Assumption 1 (informal)**.** For all states that deterministically lead to a zero-rewarding absorbing state, the MPC values can only depend on the joint reward distribution of this state's actions and on the timestep.

This is a strict requirement for backward value calculations, but also extends to more general computation schemes. Under this assumption, we prove the following result:

Claim 2.2 (informal)**.** *For any $\ell \geq 2, A \geq 2, H \geq \ell + 1$ and any MPC agent whose values satisfy Assumption 1, there exists an environment with $S \leq A^\ell + \ell + 1$ states and deterministic transitions in which the ratio between the value of the MPC agent and the optimal value is less than $A^{1-\ell/2}$. In contrast, there exist optimal MPC agents in all constructed environments.*

Most standard procedures for calculating MPC agents rely on backward induction (and follow our assumption), so they cannot be guaranteed to converge to a near-optimal MPC agent even in planning scenarios (where all distributions are known). Due to space limitations, we state the exact formulation of the controller, the assumption and the proof to Appendix I.2, and only provide intuition. We also refer to Appendix I.3 for a numerical illustration of the values in the counterexample.

Proof Sketch. The environment we consider starts at a fixed initial state and allows the agent to choose one of two paths: the first leads to a chain of states ('a line') and the second to a full tree, both of depth ℓ (see Figure 1b). Rewards are located both at the last state of the line and in all the leaves of the tree, and all these states lead to a terminal non-rewarding state. The agent gets to see the rewards only after deciding whether to traverse the tree or follow the line.

By Assumption 1, the values at both the leaves of the tree and at the end of the line only depend on the reward distribution at these states. We propose two candidates for the reward distribution to be placed in these states. The first candidate s^D deterministically assigns a reward of $A^{-\ell/2}$ to all actions, while the second s^B independently assigns a reward $\mathrm{Ber}(A^{-\ell})$ to all actions. Notably, if s^B-like states are located at all leaf states, the agent can actively navigate to realized unit rewards to collect a constant value in expectation, but at the end of the line, s^B can yield a value of at most $A^{-\ell+1}$. However, the value calculation procedure will assign the same value for s^B in both cases: we use this to trick the algorithm into always choosing the line, regardless of how it allocates the values, while ensuring that the value of the tree is always higher by a factor of $A^{\ell/2-1}$. □

3 Adaptive batching policies (ABPs)

Claim 2.1 demonstrates an important failure point in fixed batching policies: there might be critical times or states in which the lookahead information is critical, while others in which it has little benefit. In the example, the critical state is the root of the tree, while lookahead in the initial state has no significance. Fixed batching cannot adapt to such timing or states, even in simple environments, and this limitation will be even more severe in complex stochastic environments (e.g., stochastic delays). To overcome this, we propose working with a wider class of *adaptive batching policies* (ABPs). Specifically, we allow the lookahead range to depend on the state and timestep. Starting at s_1, we pick a batching horizon of $B_1 = B_1(s_1) \leq \ell$ and obtain B_1-step lookahead information. Then, we play for B_1 steps and reach state s_{B_1+1}. From there, we pick a new batching horizon $B_{B_1+1} = B_{B_1+1}(s_{B_1+1})$, act for B_{B_1+1} steps, and so forth.

Algorithm 1 Adaptive Batching Policies (ABPs)

Input: Initial state s_1
Initialize: $h = 1$ (or $\mathfrak{t}_1^\pi = 1, i = 1$)
while $h \leq H$ **do**
  Pick $B_h = B_h^\pi(s_h) \in \{1, \ldots, \ell_h\}$
  Observe $I = I_{h,B_h}(s_h)$ and set $\phi^h = \phi^h(s_h, I; \pi)$
  for $t = h, \ldots, h + B_h - 1$ **do**
    Play $a_t = \phi_t^h(s_t)$
    Collect $R_t = \mathfrak{R}_{t|h}(s_h, \phi, I)$ and transition to $s_{t+1} = \mathfrak{s}_{t+1|h}(s_h, \phi, I)$
  end for
  $h \leftarrow h + B_h$ (or $\mathfrak{t}_{i+1}^\pi \leftarrow \mathfrak{t}_i^\pi + B_{\mathfrak{t}_i^\pi}, i \leftarrow i+1$)
end while

More formally, an ABP π is specified by two mappings: i) Batching horizons $B_h = B_h^\pi(s) \in [\ell_h]$ that specify the utilized lookahead range when starting a new batch in step h and state s, and ii)

Deterministic policies $\phi^h = \phi^h(s, I; \pi) \in \Pi_{det}$ that determine how the agent will act from step h to step $h + B_h - 1$ given the lookahead information.

Notably, when the transitions are stochastic, the starting times of new batches may become random; we denote by $\mathfrak{t}_i^\pi$ the (random) step in which a policy π started its i^{th} batch, and by E_h^n, the event in which some batch started at step h. In particular, we always start a new batch at the first step, and so $\mathfrak{t}_1^\pi = 1$ and E_1^n always holds. Then, in step $h = \mathfrak{t}_i^\pi$, the ABP picks $B_h = B_h^\pi(s_h)$ and observes $I_h = I_{h,B_h}(s_h)$ (and *not* $I_{h,\ell}(s_h)$[2]). The ABP plays ϕ^h for B_h steps, ending in $s_{h+B_h} = \mathfrak{s}_{h+B_h|h}(s_h, \phi^h, I_h)$, and the next batch then starts at $\mathfrak{t}_{i+1}^\pi = \mathfrak{t}_i^\pi + B_h$ in state $s_{\mathfrak{t}_{i+1}^\pi} = s_{h+B_h}$. The process continues until H steps have been played. For clarity, we also describe the interaction protocol in Algorithm 1. We denote the set of policies that abide by this protocol as Π_B.

Remark 1. *Since the lookahead contains perfect information on the transition realization, a deterministic policy is completely equivalent to pre-choosing a sequence of B_h actions (and this is how transition lookahead is defined in [19]). Nonetheless, we still define the problem with Markov policies to facilitate future extensions to imperfect lookahead information.*

Going back to the example of Claim 2.1, even if the delay from s_1 to the root of the tree s_r is stochastic, an ABP could look one step ahead at a time by setting $B_h^\pi(s_1) = 1$, waiting to reach s_r, and only then utilize the full lookahead ($B_h^\pi(s_r) = \ell$) to optimally collect the rewards at the leaves. Similarly, it is also easy to verify that an ABP can optimally solve the tree-and-line example in Claim 2.2; the optimal solution will choose $B_1 = 1$ and then continue until the end with $B_2 = \ell$.

Values and performance criterion. We define the value of an ABP π upon starting a new batch at step h and state s to be $V_h^\pi(s) = \mathbb{E}\Big[\sum_{t=h}^H R_t | s_h = s, E_h^n, \pi\Big]$, and the optimal value of ABPs $V_1^*(s) = \max_{\pi \in \Pi_B} V_1^\pi(s)$. Importantly, since a new batch starts at step h, there is no overlap between lookahead information obtained before step h and any event after step h (except for s_h, on which we condition). Therefore, the value will be the same regardless of the path of the agent on its way to a new batch at $s_h = s$.

In learning scenarios, an agent interacts with the environment for K episodes, aiming to compete with the best ABP. The initial state in episode k is s_1^k, and we allow it to be arbitrarily chosen by an adversary. Denoting by π^k the policy of the agent in episode k, we measure agents by their regret, defined as the cumulative difference between the value of an optimal ABP and their expected cumulative reward: $\mathrm{Reg}(K) = \sum_{k=1}^K \Big(V_1^*(s_1^k) - V_1^{\pi^k}(s_1^k)\Big)$.

3.1 Optimal Planning

We start with the case where the model of the environment is perfectly known, with the goal of determining the optimal ABP and its value. In Proposition 3.1, we derive the dynamic programming equations that characterize the optimal value of ABPs and yield the optimal policies. Let $\mathfrak{Q}_h^*(s, B; I, V)$ be the optimal reward gained starting from step h and state s, given B-step lookahead I and assuming a value $V \in \mathbb{R}^{\mathcal{S}}$ at the end of the lookahead range

$$\mathfrak{Q}_h^*(s, B; I, V) = \max_{\phi \in \Pi_{det}} \left\{ \sum_{t=h}^{h+B-1} \mathfrak{R}_{t|h}(s, \phi, I) + V(\mathfrak{s}_{h+B|h}(s, \phi, I)) \right\}. \tag{2}$$

Then, the following holds:

Proposition 3.1. *There exists an optimal adaptive batching policy π^* that maximizes the value $V_1^\pi(s)$ simultaneously for all $s \in \mathcal{S}$. For any $h \in [H]$ and $s \in \mathcal{S}$, the optimal values are given by the dynamic programming equation*

$$V_h^*(s) = \max_{B \in [\ell_h]} \mathbb{E}_{I \sim \mathcal{I}_{h,B}(s)} \big[\mathfrak{Q}_h^*(s, B; I, V_{h+B}^*)\big], \tag{3}$$

where $V_{H+1}^*(s) = 0$ for all $s \in \mathcal{S}$ and the expectation samples fresh lookahead information independently between timesteps. An optimal policy chooses lookahead ranges according to

$$B_h^*(s) \in \arg\max_{B \in [\ell_h]} \mathbb{E}_{I \sim \mathcal{I}_{h,B}(s)} \big[\mathfrak{Q}_h^*(s, B; I, V_{h+B}^*)\big]. \tag{4}$$

[2]We discuss the implications of this feedback in Section 4.

Finally, when starting a new batch at step h and state s of length $B_h^* = B_h^*(s)$ and with lookahead information $I = I_{h,B_h^*}(s)$, it is optimal to play the deterministic policy

$$\phi^{*,h} \in \arg\max_{\phi \in \Pi_{det}} \left\{ \sum_{b=0}^{B_h^*-1} \mathfrak{R}_{h+b|h}(s,\phi,I) + V^*_{h+B_h^*}\Big(\mathfrak{s}_{h+B_h^*|h}(s,\phi,I)\Big) \right\}. \tag{5}$$

The proof can be found in Appendix B. Inspired by [16] in the case of one-step lookahead, we design an augmented Markov environment $\mathcal{M}^{\mathrm{B}}$ in which the state includes both the lookahead information and the number of steps until the current batch ends. The action space is adapted to also include the choice of the batching horizon, in steps where it is relevant. We explain why this construction is almost equivalent to ABPs in the original MDP, with one delicate change: Markov policies in $\mathcal{M}^{\mathrm{B}}$ may also depend on the number of past batches (while ABPs do not use this as input). We show that the optimal policy does not use this extra information and results in being an ABP. We also rely on this construction to derive the value of arbitrary ABPs (Corollary 2 in the appendix).

Efficient Value Computation. We now briefly discuss how to efficiently solve the dynamic programming equations of ABPs. Starting from a step h and a state s, let $\mathfrak{J}^I_{h,B}(s,s')$ be the maximal cumulative reward after playing B steps, given lookahead information I and under the constraint that the environment arrives at state s' on step $h+B$:

$$\mathfrak{J}^I_{h,B}(s,s') = \max_{\phi \in \Pi_{det}: \mathfrak{s}_{h+B|h}(s,\phi,I)=s'} \left\{ \sum_{t=h}^{h+B-1} \mathfrak{R}_{t|h}(s,\phi,I) \right\}. \tag{6}$$

In addition, let $\mathfrak{S}^I_{h,B}(s)$ be the set of reachable states after B steps

$$\mathfrak{S}^I_{h,B}(s) = \big\{ s' \in \mathcal{S} : \exists \phi \in \Pi_{det},\ s.t.\ \mathfrak{s}_{h+B|h}(s,\phi,I) = s' \big\}. \tag{7}$$

Using standard dynamic programming arguments, we have

$$\mathfrak{Q}^*_h(s,B;I,V) = \max_{s' \in \mathfrak{S}^I_{h,B}(s)} \big\{ \mathfrak{J}^I_{h,B}(s,s') + V(s') \big\}, \tag{8}$$

and given lookahead information, both $\mathfrak{J}$ and $\mathfrak{S}$ can be updated recursively by[3]

$$\begin{aligned}
\mathfrak{J}^I_{h,B+1}(s,s') &= \max_{\substack{\tilde{s} \in \mathfrak{S}^I_{h,B}(s), a \in \mathcal{A}: \\ \mathfrak{s}_{h+B+1|h+B}(\tilde{s},a,I)=s'}} \big\{ \mathfrak{J}^I_{h,B}(s,\tilde{s}) + \mathfrak{R}_{h+B|h+B}(\tilde{s},a,I) \big\}, \quad \text{and} \\
\mathfrak{S}^I_{h,B+1}(s) &= \big\{ s' \in \mathcal{S} : \exists \tilde{s} \in \mathfrak{S}^I_{h,B}(s), a \in \mathcal{A},\ s.t.\ \mathfrak{s}_{h+B+1|h+B}(\tilde{s},a,I) = s' \big\},
\end{aligned} \tag{9}$$

with the initialization $\mathfrak{J}^I_{h,0}(s,s) = 0$ and $\mathfrak{S}^I_{h,0}(s) = \{s\}$. If s' is not reachable from s, we arbitrarily define $\mathfrak{J}^I_{h,B+1}(s,s') = 0$. For any step h state s and lookahead information I, the computation complexity of these calculations is $\mathcal{O}(SA\ell)$, and the memory complexity is $\mathcal{O}(S\ell)$. Therefore, we can efficiently compute $\mathfrak{Q}^*_h(s,B;I,V^*_{h+B})$ in Equations (3) and (4), and only need to approximate the expectation w.r.t. the lookahead information. Luckily, since rewards are bounded, all quantities concentrate well, and one could sample $poly\big(1/\epsilon, S, H, \log\frac{1}{\delta}\big)$ lookahead realizations to estimate the values up to an ϵ-accuracy w.h.p. using this procedure. Finally, we remark that given B_h and I_h, after calculating $\mathfrak{J}$ and $\mathfrak{S}$ and solving for the state s_{h+B_h}, the optimal policy $\phi^{*,h}$ can be calculated by backtracking from the final state to the initial state $s_h = s$.

4 Learning the Optimal ABP

Finally, we address the problem of learning to interact with unknown environments. We study episodic settings and assume that the interaction lasts for K episodes. Following the ABP protocol, each episode is divided into batches determined by the agent. Starting from an arbitrary state s_1^k, the agent picks the first batching horizon B_1^k and observes B_1^k-step lookahead information I_h^k containing both realized rewards and transitions. Then, it plays for B_1^k steps according to a policy $\phi^{k,h}$ until

[3] For brevity, we slightly abuse the use of notations $\mathfrak{s}, \mathfrak{R}$ and input the lookahead information in step h; notice that it directly determines the relevant state/rewards in step $h+B$.

Algorithm 2 Adaptive Lookahead Upper Confidence Bound (AL-UCB)

for $k = 1, 2, ...$ **do**
  Initialize $\bar{V}^k_{H+1}(s) = 0$ # Planning
  for $h = H, H-1, .., 1$ and $s \in \mathcal{S}$ **do**
    Calculate $\bar{Q}^k_h(s, B)$ via eq. (11) for all $B \in [\ell_h]$ and set $\bar{V}^k_h(s) = \max_{B\in[\ell_h]}\{\bar{Q}^k_h(s, B)\}$
  end for
  Start from $h = 1$ in state s^k_1 # Acting
  while $h \le H$ **do**
    Pick $B^k_h \in \arg\max_{B\in[\ell_h]}\{\bar{Q}^k_h(s^k_h, B)\}$; observe B^k_h-step lookahead I^k_h from state s^k_h
    Play for B^k_h steps a policy $\phi^{k,h}$ as specified in eq. (12) and update $h \leftarrow h + B^k_h$
  end while
  Update the empirical estimators and counts for all s^k_h in which a new batch started and the respective batching horizons
end for

reaching the starting point of the second batch $\mathfrak{t}^k_2 = \mathfrak{t}^k_1 + B^k_1$ (with $\mathfrak{t}^k_1 = 1$), and so forth until H steps were played. We emphasize that we assume much weaker feedback than the original ℓ-step lookahead: first, the lookahead is only observed at the beginning of batches (and not in every step), and second, the observed lookahead range is $B^k_h \le \ell$. This can be very beneficial when the lookahead information is costly – whether the cost is computational or the agent must pay to acquire it.

We continue and define several useful empirical quantities. Let $\mathcal{K}^k_h(s, B) \subseteq [k-1]$ be all episodes up to $k-1$ (inclusive) in which the agent started a new batch in state s at step h and asked to observe a lookahead range of B. The number of such episodes is denoted by $n^k_h(s, B) = \left|\mathcal{K}^k_h(s, B)\right|$. Then, we denote the empirical expectation w.r.t. the collected lookahead data of an arbitrary function f by

$$\widehat{\mathbb{E}}^k_{h,s,B}[f(I)] = \frac{1}{n^k_h(s, B) \vee 1} \sum_{m\in\mathcal{K}^k_h(s,B)} f(I^m_h(s)),$$

where $x \vee y = \max\{x, y\}$. We remark that as long as computing $f(I)$ is tractable, calculating this expectation requires at most k computations. We define the empirical variance of the Q-values to be

$$\widehat{\mathrm{Var}}^k_h(s, B; V) = \widehat{\mathbb{E}}^k_{h,s,B}\left[\mathfrak{Q}^*_h(s, B; I, V)^2\right] - \left(\widehat{\mathbb{E}}^k_{h,s,B}[\mathfrak{Q}^*_h(s, B; I, V)]\right)^2, \tag{10}$$

where $\mathfrak{Q}^*$ is defined in eq. (2).

Using these notations, we are ready to present our algorithm, which we call Adaptive Lookahead Upper Confidence Bound (AL-UCB). We adapt the optimistic algorithms for RL, and, in particular, the one-step lookahead algorithms in [16]. In each episode, the algorithm recursively calculates optimistic values $\bar{V}^k_h(s)$ and utilizes them to determine the batching horizons B^k_h and the policies inside batches $\phi^{k,h}$. The optimism is achieved via variance-based bonuses of the form

$$b^k_h(s, B) = \tilde{\Theta}\left(\sqrt{\frac{\widehat{\mathrm{Var}}^k_h(s, B; \bar{V}^k_{h+B})}{n^k_h(s, B)}} + \frac{1}{n^k_h(s, B)}\right)$$

for any $h \in [H], s \in \mathcal{S}$ and $B \in [\ell_h]$ (see eq. (18) in the appendix for the precise expression). Then, given the bonuses, the optimistic Q-values are defined as

$$\bar{Q}^k_h(s, B) = \min\left\{\widehat{\mathbb{E}}^k_{h,s,B}\left[\mathfrak{Q}^*_h(s, B; I, \bar{V}^k_{h+B})\right] + b^k_h(s, B), H - h + 1\right\}. \tag{11}$$

Notice that we truncate the values to ensure they stay in the domain $[0, H-h+1]$. The optimistic value is then defined as the maximum over all potential batching horizons $\bar{V}^k_h(s) = \max_{B\in[\ell_h]}\{\bar{Q}^k_h(s, B)\}$.

Having calculated the values, the algorithm repeatedly selects batching horizons that maximize the optimistic Q-values $B^k_h \in \arg\max_{B\in[\ell_h]}\{\bar{Q}^k_h(s, B)\}$, and then, upon observing I^k_h, executes the greedy Markov policy that maximizes the reward given the lookahead and the optimistic value at the end of the lookahead

$$\phi^{k,h} \in \arg\max_{\phi\in\Pi_{det}}\left\{\sum_{t=h}^{h+B^k_h-1} \mathfrak{R}_{t|h}(s^k_h, \phi, I^k_h) + \bar{V}^k_{h+B^k_h}(\mathfrak{s}_{h+B^k_h|h}(s^k_h, \phi, I^k_h))\right\}. \tag{12}$$

We remark that our discussion on efficient calculations in Section 3 also applies here: one could use quantities similar to $\mathfrak{J}, \mathfrak{S}$ to efficiently find $\phi^{k,h}$ (see Appendix F for further details). We also note that while the bonuses affect the optimistic values $\bar{Q}_h^k(s,B), \bar{V}_{h+B_h^k}^k$, they do not directly appear in the policy $\phi^{k,h}$: this is because the lookahead information removes all uncertainty inside the batch. In other words, inside a batch, the algorithm is completely greedy w.r.t. the lookahead information and optimistic values. The full description of AL-UCB can be found in Algorithm 2. Our algorithm enjoys the following regret guarantees:

Theorem 4.1. *For any $\delta \in (0,1)$, w.p. at least $1-\delta$ the regret of AL-UCB is upper bounded by* $\mathrm{Reg}(K) = \mathcal{O}\Big(\sqrt{H^3SK\ell}\ln\frac{SH\ell K}{\delta} + H^3S^2\ell\ln^2\frac{SH\ell K}{\delta}\Big)$.

One major challenge that also appeared in the one-step lookahead [16] is that lookahead exponentially increases the effective state-space. Specifically, for some empirical transition estimate $\hat{P}_h^k$, previous works often require bounding a (lower-order) term of the form $\left(\hat{P}_h^k - P_h\right)^T\left(\bar{V}_{h+1}^k - V_{h+1}^*\right)$, and one standard approach yields a bound of the form (see, e.g., [9]):

$$\sum_{s'}\hat{P}_h^k(s'|s,a)\big(\bar{V}_{h+1}^k(s') - V_{h+1}^*(s')\big) \lesssim \left(1+\frac{1}{H}\right)\sum_{s'}P_h(s'|s,a)\big(\bar{V}_{h+1}^k(s') - V_{h+1}^*(s')\big) + \tilde{\mathcal{O}}\left(\frac{SH}{n_h^k(s,a)}\right). \tag{13}$$

However, this bound depends on S, and when generalizing this to incorporate lookahead by augmenting it to the state space, the relevant distribution is the lookahead distribution $\mathcal{I}_{h,B}(s)$. Thus, instead of S, one would get the support of this distribution, which has an exponential nature due to its transition elements and continuous components due to reward lookahead. With one-step reward lookahead, Merlis [16] tackled this issue by using uniform concentration on the value, suffering a factor of $\sqrt{A}$ (which did not affect the final rate due to the transition estimation terms). In our case, the 'effective action' would be to choose the state at the end of the lookahead, which would lead to a regret degradation of $\sqrt{S}$. On the other hand, with transition lookahead, Merlis [16] reformulated the probability space using a special list structure of the optimal policy, so that it becomes polynomial. This structure does not hold in our case (even with one-step reward and transition lookahead).

Our way of solving this is to prove uniform concentration on the problematic error term using Freedman's inequality, alongside covering arguments that only lead to a polynomial degradation in lower-order terms. Specifically, we show that w.h.p.,

$$\widehat{\mathbb{E}}_{h,s,B}^k\left[\mathfrak{Q}_h^*(s,B;I,\bar{V}_{h+B}^k) - \mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)\right] \le \left(1+\frac{1}{2H}\right)\mathbb{E}_{I\sim\mathcal{I}}\left[\mathfrak{Q}_h^*(s,B;I,\bar{V}_{h+B}^k) - \mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)\right] + \mathcal{O}\left(\frac{H^2SL_\delta^k}{n_h^k(s,B)}\right),$$

which is the equivalent of eq. (13) in our setting; notice that this inequality depends on the size of the original state space S, and not that of an augmented space. The aforementioned event, with the exact constants, is formally defined in Appendix D.1 as $E^{Err}(k)$, and the general concentration claim used to bound its probability is proved in Lemma 8. This result allows us to propagate errors from the beginning of a batch at step h to the beginning of the next batch at step $h+B$, and the multiplicative degradation due to this error propagation is at most $\left(1+\frac{1}{2H}\right)^H = \mathcal{O}(1)$. The full proof includes other technical challenges related to adapting to the batch structure and can be found in Appendix E. We remark that our approach can be applied to the one-step transition lookahead to greatly simplify the proof in [16], obviating the need for the list-based manipulations.

Comparison to one-step lookahead. The minimax optimal regret rates with either one-step reward or transition lookahead information are $\tilde{\mathcal{O}}\left(\sqrt{H^3SAK}\right)$ and $\tilde{\mathcal{O}}\left(\sqrt{H^2SK}(\sqrt{H}+\sqrt{A})\right)$, respectively [16]. Limiting our results to $\ell = 1$ (where ABPs are equivalent to a general lookahead policy), we show an improved regret rate of $\tilde{\mathcal{O}}\left(\sqrt{H^3SK}\right)$, removing all dependence on the size of the action space. This is not surprising: in any visited state, we observe all possible actions before acting. Nonetheless, as previously explained, it is not clear whether the proof in [16] can be adapted to achieve this rate. As for lower bounds, following a similar reduction to a bandit-like setting via trees

[6], this rate is tight. Specifically, taking the example in [6] and removing all actions but one in all the leaves would remove all benefits of one-step lookahead information, so that lookahead algorithms get the same information as vanilla RL algorithms. Removing these actions reduces the standard RL lower bound by a factor of $\sqrt{A}$: to a value of $\Omega\left(\sqrt{H^3SK}\right)$.

Tightness for $\ell > 1$. For multi-step lookahead, it is unclear whether the bounds are tight in the maximal lookahead ℓ. One simple way to reduce the feedback from ℓ-step lookahead to standard RL is to add delays: force the environment to progress only in steps where $h = n\ell$ and freeze it (with no reward) otherwise. This can be achieved by duplicating each state ℓ times. Equivalently, we can transform an ℓ-step lookahead environment with ℓ delays to a one-step lookahead environment with S/ℓ states. Based on this principle (and specifically, duplicating the leaves in the standard tree-like construction in [6]), the immediate candidate for a lower bound is $\Omega\left(\sqrt{H^3SK/\ell}\right)$: different by a factor of ℓ from our upper bound. For most applications, it is reasonable to treat ℓ as an absolute constant (and, in particular, $\ell \ll H$), and so this gap is not major. Moreover, part of this gap is very likely due to our feedback model: the fact that at batch starts, we observe B_h-step lookahead and not ℓ_h-step lookahead. In fact, were we to observe ℓ_h-step lookahead at batch starting points, we could update the observations simultaneously for any $B \in [\ell_h]$, and all dependencies in $n_h^k(s, B)$ would be replaced by $n_h^k(s)$. In particular, this would lead to a regret upper bound of $\tilde{\mathcal{O}}\left(\sqrt{H^3SK}\right)$ for any ℓ (see Appendix E.1 for more details). Similarly, looking at fixed batching policies and forcing B_h to be independent of s (i.e., to be any fixed function of h), we would also get a predefined batching horizon for any timestep. Therefore, we could maintain a single count $n_h^k(s)$. Then, following the same analysis would immediately lead to a regret of $\tilde{\mathcal{O}}\left(\sqrt{H^3SK}\right)$ compared to the best policy with the same fixed batching profile.

We hypothesize that for large enough ℓ, the regret should diminish, since the feedback becomes increasingly similar to full information, but leave this for future work.

5 Conclusions and Future Work

In this work, we tackled the problem of planning and learning with multiple steps of lookahead information. We demonstrated the failure of fixed batching policies and the exponential suboptimality of common value calculations for MPC schemes, and proposed focusing on adaptive batching policies: both improving fixed batching policies while retaining the ability to compute the optimal strategy. We derived the optimal Bellman equations for ABPs and showed how to learn the optimal ABP when interacting with unknown tabular episodic environments. Our regret bounds are near-optimal up to factors that depend on the lookahead horizon ℓ, which is typically treated as a small constant.

More broadly, the aim of the work is to expand the family of tractable strategies with multi-step lookahead and show how to conduct learning in this scenario. However, there is still much to be done to fully characterize the tractable regimes in this setting. In particular, though it has been proven that exactly computing the optimal lookahead policy is hard [19] and there are exponential-in-ℓ approximation schemes [20] for similar settings, it remains open whether there exist polynomial planning schemes with provable approximation guarantees. Establishing such results falls beyond the scope of our work. Alternatively, after showing the failure of backwards-induced MPC schemes, it would be interesting to study whether there are tractable global approaches for computing better values. We conjecture that calculating the optimal values is NP-hard, since they may depend delicately on the entire structure of the environment.

Other directions involve closing the regret gap or studying algorithmic alternatives that may be better suited for practical deployment. For example, instead of solving a planning problem inside each batch, one could use rollouts; the model-based optimistic value calculations could be replaced by model-free approaches; and so forth. After pursuing these directions, one could adapt the resulting algorithms to deep RL and apply them to some of the many applications in which lookahead information is available. Finally, our work assumes complete and perfect ℓ-step information. In practice, information may be distorted (especially further into the future), and some future information might be missing. We believe our analysis can be naturally combined with that of [15] to incorporate imperfect future predictions, but leave this for future study.

Acknowledgments

This research was supported by Israel Science Foundation research grant (ISF's No. 4118/25) and the Maimonides Fund's Future Scientists Center.

References

[1] Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, pages 263–272. PMLR, 2017.

[2] Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandit algorithms with supervised learning guarantees. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 19–26, 2011.

[3] Eduardo F Camacho and Carlos Bordons. *Model predictive control*. Springer, 2007.

[4] Stephen Chung, Ivan Anokhin, and David Krueger. Thinker: Learning to plan and act. *Advances in Neural Information Processing Systems*, 36:22896–22933, 2023.

[5] Christoph Dann, Lihong Li, Wei Wei, and Emma Brunskill. Policy certificates: Towards accountable reinforcement learning. In *International Conference on Machine Learning*, pages 1507–1516, 2019.

[6] Omar Darwiche Domingues, Pierre Ménard, Emilie Kaufmann, and Michal Valko. Episodic reinforcement learning in finite mdps: Minimax lower bounds revisited. In *Algorithmic Learning Theory*, pages 578–598. PMLR, 2021.

[7] Yonathan Efroni, Nadav Merlis, Mohammad Ghavamzadeh, and Shie Mannor. Tight regret bounds for model-based reinforcement learning with greedy policies. In *Advances in Neural Information Processing Systems*, pages 12224–12234, 2019.

[8] Yonathan Efroni, Mohammad Ghavamzadeh, and Shie Mannor. Online planning with lookahead policies. *Advances in Neural Information Processing Systems*, 33:14024–14033, 2020.

[9] Yonathan Efroni, Nadav Merlis, Aadirupa Saha, and Shie Mannor. Confidence-budget matching for sequential budgeted learning. In *International Conference on Machine Learning*, pages 2937–2947. PMLR, 2021.

[10] Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(Apr):1563–1600, 2010.

[11] Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. Is q-learning provably efficient? *Advances in Neural Information Processing Systems*, 2018:4863–4873, 2018.

[12] Yingying Li, Xin Chen, and Na Li. Online optimal control with linear dynamics and predictions: Algorithms and regret analysis. *Advances in Neural Information Processing Systems*, 32, 2019.

[13] Yiheng Lin, Yang Hu, Guanya Shi, Haoyuan Sun, Guannan Qu, and Adam Wierman. Perturbation-based regret analysis of predictive control in linear time varying systems. *Advances in Neural Information Processing Systems*, 34:5174–5185, 2021.

[14] Yiheng Lin, Yang Hu, Guannan Qu, Tongxin Li, and Adam Wierman. Bounded-regret mpc via perturbation analysis: Prediction error, constraints, and nonlinearity. *Advances in Neural Information Processing Systems*, 35:36174–36187, 2022.

[15] Chenbei Lu, Zaiwei Chen, Tongxin Li, Chenye Wu, and Adam Wierman. Reinforcement learning with imperfect transition predictions: A bellman-jensen approach. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[16] Nadav Merlis. Reinforcement learning with lookahead information. *Advances in Neural Information Processing Systems*, 37:64523–64581, 2024.

[17] Nadav Merlis, Dorian Baudry, and Vianney Perchet. The value of reward lookahead in reinforcement learning. *Advances in Neural Information Processing Systems*, 37:83627–83664, 2024.

[18] Thomas M Moerland, Anna Deichler, Simone Baldi, Joost Broekens, and Catholijn M Jonker. Think neither too fast nor too slow: The computational trade-off between planning and reinforcement learning. In *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS), Nancy, France*, pages 16–20, 2020.

[19] Corentin Pla, Hugo Richard, Marc Abeille, Nadav Merlis, and Vianney Perchet. On the hardness of reinforcement learning with transition look-ahead. In *Proceedings of The 29th International Conference on Artificial Intelligence and Statistics*, pages 55–63, 2026.

[20] Corentin Pla, Hugo Richard, Marc Abeille, and Vianney Perchet. Near-optimal reinforcement learning with multi-step transition lookahead. *arXiv preprint arXiv:2609.11807*, 2026.

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

[22] Aviv Rosenberg, Assaf Hallak, Shie Mannor, Gal Chechik, and Gal Dalal. Planning and learning with adaptive lookahead. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 9606–9613, 2023.

[23] Max Simchowitz and Kevin G Jamieson. Non-asymptotic gap-dependent regret bounds for tabular mdps. In *Advances in Neural Information Processing Systems*, pages 1153–1162, 2019.

[24] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[25] Aviv Tamar, Garrett Thomas, Tianhao Zhang, Sergey Levine, and Pieter Abbeel. Learning from the hindsight plan—episodic mpc improvement. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 336–343. IEEE, 2017.

[26] Chenkai Yu, Guanya Shi, Soon-Jo Chung, Yisong Yue, and Adam Wierman. The power of predictions in online control. *Advances in Neural Information Processing Systems*, 33: 1994–2004, 2020.

[27] Andrea Zanette and Emma Brunskill. Tighter problem-dependent regret bounds in reinforcement learning without domain knowledge using value function bounds. In *International Conference on Machine Learning*, pages 7304–7312. PMLR, 2019.

[28] Runyu Zhang, Yingying Li, and Na Li. On the regret analysis of online lqr control with predictions. In *2021 American Control Conference (ACC)*, pages 697–703. IEEE, 2021.

[29] Zihan Zhang, Xiangyang Ji, and Simon Du. Is reinforcement learning more difficult than bandits? a near-optimal algorithm escaping the curse of horizon. In *Conference on Learning Theory*, pages 4528–4531. PMLR, 2021.

[30] Zihan Zhang, Yuxin Chen, Jason D Lee, and Simon S Du. Settling the sample complexity of online reinforcement learning. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 5213–5219. PMLR, 2024.

[31] Ziyi Zhang, Yorie Nakahira, and Guannan Qu. Predictive control and regret analysis of non-stationary MDP with look-ahead information. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856.

Table of Contents

A Data Generation Process **14**

B Description of an Equivalent Augmented MDP **15**

C Additional Notations **21**

D Good Events **22**
D.1 The First Good Event – Concentration 22
D.2 Optimism Under the First Good Event 25
D.3 The Second Good Event – Cumulative Concentration 26

E Regret Analysis **28**
E.1 Regret When Lookahead Is Obtained for All Batching Horizons 32
E.2 Lemmas for Bounding Variance Terms 33

F Computational Complexity **35**

G Concentration Results **36**

H Existing Auxiliary Lemmas **38**

I Failure of Existing Approaches **40**
I.1 Fixed Batching Policies 40
I.2 Model Predictive Control 40
I.3 Numerical Illustrations 42

J Table of Notations **43**

A Data Generation Process

We first specify in detail how lookahead information is generated. While there are other equivalent ways to generate it, this generation process will simplify the analysis and, in particular, will trivially imply equivalence between obtaining lookahead information and interacting with an augmented environment described in Appendix B.

The key observation on the adaptive batching policy space is that the lookahead information $I_{h,B_h}(s_h)$ is only observed at the beginning of a batch, after choosing a new batch length B_h, and even if obtained, additional lookahead information is not used in the middle of the batch. Therefore, for ABPs, we can assume w.l.o.g. that the lookahead information $I_{h,B_h}(s_h)$ is independently generated only *after reaching state s_h and picking a lookahead range B_h*.

Formally, for any $h \in [H]$ and $B \in \{1, \ldots, \ell_h\}$, denote by $\boldsymbol{s}'_{h,B} \in \mathcal{S}^{SA\ell}$, the B-step transition lookahead information starting from step h. Specifically, for any $h \le t \le h+B-1$, any reachable $s \in \mathcal{S}$ and any $a \in \mathcal{A}$, upon playing a at step t and state s, the agent would transition to $s_{t+1} = \boldsymbol{s}'_{h,B}(s,a,t)$. To keep the dimension requirements, for $t \in \{h+B, \ldots, h+\ell-1\}$, we pad with $\boldsymbol{s}'_{h,B}(s,a,t) = \emptyset$ (for some arbitrary unreachable state $\emptyset$), and also define $\boldsymbol{s}'_{h,0} = \emptyset^{\otimes SA\ell}$. We similarly denote reward lookahead by $\boldsymbol{R}_{h,B} \in \mathbb{R}^{SA\ell}$, so that upon playing action a at step $t \in \{h, \ldots, h+B-1\}$ and state s, the agent would collect a reward $R_t = \boldsymbol{R}_{h,B}(s,a,t)$, and set $\boldsymbol{R}_{h,B}(s,a,t) = 0$ for $t > h+B-1$. We again denote $\boldsymbol{R}_{h,0} = 0^{\otimes SA\ell}$.

Lookahead Sampling Procedure. We further specify the lookahead sampling procedure to ensure equivalence between $I_{h,B_h}(s_h)$ and the observations $\boldsymbol{R}_{h,B}, \boldsymbol{s}'_{h,B}$ sampled from $s_h = s$. The procedure consists of two stages.

1. *Sampling*. Given h, B for all $t \in \{h, \ldots, h+B-1\}$, sample $\boldsymbol{R}_{h,B}(s,a,t)$ and $\boldsymbol{s}'_{h,B}(s,a,t)$ for all $s \in \mathcal{S}, a \in \mathcal{A}$. The samples can be correlated between states and actions (if the reward and/or transitions are correlated), but are sampled independently between different values of t. This sampling procedure gives lookahead information $\boldsymbol{R}_{h,B}, \boldsymbol{s}'_{h,B}$ also on unreachable states.
2. *Pruning*. To make the information equivalent to the lookahead information $I_{h,B_h}(s_h)$, we need to remove any information on unreachable states; we do it recursively. First, for $t = h$, the only reachable state is s_h. Therefore, for any $s \neq s_h$, we set $\boldsymbol{R}_{h,B}(s,a,h) = 0$ and $\boldsymbol{s}'_{h,B}(s,a,h) = \emptyset$. The set of reachable states in step $h+1$ is

$$\mathfrak{S}_{h,1}(s_h, \boldsymbol{s}'_{h,B}) = \left\{ s' \in \mathcal{S} : \exists a \in \mathcal{A} \;\; s.t. \;\; \boldsymbol{s}'_{h,B}(s_h,a,h) = s' \right\}.$$

   Then, for any step $h+b$ (with $b \in [B-1]$), if $s \notin \mathfrak{S}_{h,b}(s_h, \boldsymbol{s}'_{h,B})$, we set $\boldsymbol{R}_{h,B}(s,a,h+b) = 0$ and $\boldsymbol{s}'_{h,B}(s,a,h+b) = \emptyset$ for all $a \in \mathcal{A}$. The reachable states in step $h+b+1$ will now be

$$\mathfrak{S}_{h,b+1}(s_h, \boldsymbol{s}'_{h,B}) = \left\{ s' \in \mathcal{S} : \exists s \in \mathfrak{S}_{h,b}(s_h, \boldsymbol{s}'_{h,B}), a \in \mathcal{A} \;\; s.t. \;\; \boldsymbol{s}'_{h,B}(s,a,h+b) = s' \right\}.$$

We write $\boldsymbol{s}'_{h,B} \sim \mathcal{P}_h^B(s)$ and $\boldsymbol{R}_{h,B} \sim \mathcal{R}_h^B(s)$ to represent the independent sampling from the aforementioned distribution starting at $s_h = s$. By construction, the information left after pruning contains exactly all rewards and transitions along all possible trajectories/policies: the same information as in $I_{h,B_h}(s_h)$. With slight abuse of notation, given a deterministic policy ϕ, we again use the notations $\mathfrak{R}_{h+b|h}(s, \phi, \boldsymbol{R}, \boldsymbol{s}')$ and $\mathfrak{s}_{h+b|h}(s, \phi, \boldsymbol{R}, \boldsymbol{s}')$ to denote the rewards and states reached from state s by playing policy ϕ on the lookahead $\boldsymbol{R}, \boldsymbol{s}'$. When the batching horizon B is clear from the context, we sometimes write $\mathcal{I}_h(s)$.

Remark 2. *The pruning step is not strictly needed for planning or when analyzing the value of policies: the pruned states can never be visited, so they cannot affect the optimal policy. Moreover, every policy that uses their information can be converted to a policy of the same value that does not use this information by sampling the information on the pruned states directly from the MDP distribution and acting as if this is the realization. Therefore, we prune the rewards mainly to make the equivalence between the lookahead information and the augmented MDP clear. On the other hand, in learning situations, observing information in unreachable states would effectively convert the bandit feedback to full information, thus making the problem much easier; we do not analyze this feedback in our paper.*

B Description of an Equivalent Augmented MDP

Having reformulated the lookahead information in a vectorized form in Appendix A, we now present an equivalent environment $\mathcal{M}^{\mathrm{B}}$, in which this information is augmented into the state. This augmentation renders the environment Markov and enables us to analyze it using standard planning tools. We call $\mathcal{M}$ the original MDP and $\mathcal{M}^{\mathrm{B}}$ the batched MDP. The different components of the batched MDP $\mathcal{M}^{\mathrm{B}}$ are defined as follows:

- **State Space:** the state is represented by a tuple $(s, \boldsymbol{R}, \boldsymbol{s}', h, B)$, where $s \in \mathcal{S}$ is the current state of the agent, $\boldsymbol{R} \in \mathbb{R}^{SA\ell}$ and $\boldsymbol{s}' \in \mathcal{S}^{SA\ell}$ encode the lookahead information, $h \in [H]$ is the time step at the original MDP, and $B \in \{0, \dots, \ell_h\}$ is the remaining steps until the end of a batch.
- **Action Space:** the agent picks an action $(a, B') \in \mathcal{A} \times [\ell_h]$, where the first component represents a real interaction and the second represents a batching horizon.
- **Initial state:** Assuming the original MDP starts at state s, the batched environment starts at state

$$(s_1, \boldsymbol{R}_1, \boldsymbol{s}'_1, h_1, B_1) \leftarrow \big(s, 0^{\otimes SA\ell}, \emptyset^{\otimes SA\ell}, 1, 0\big).$$

- **Interaction Protocol:** We denote the time index in $\mathcal{M}^{\mathrm{B}}$ by t, which will advance at a slower pace than the time index of the original MDP h. The interaction protocol is divided into the following scenarios:
  - If $h_t \le H$ and $B_t = 0$, we are about to start a new batch at timestep h_t. The agent picks the batch size using B'_t (and the action a_t has no effect). The reward is zero, and the agent transitions to the state

$$(s_{t+1}, \boldsymbol{R}_{t+1}, \boldsymbol{s}'_{t+1}, h_{t+1}, B_{t+1}) \leftarrow (s_t, \boldsymbol{R}_{h_t, B'_t}, \boldsymbol{s}'_{h_t, B'_t}, h_t, B'_t),$$

    where both $\boldsymbol{R}_{h_t, B'_t} \sim \mathcal{R}_{h_t}^{B'_t}(s_t)$ and $\boldsymbol{s}'_{h_t, B'_t} \sim \mathcal{P}_{h_t}^{B'_t}(s_t)$ are lookahead information sampled according to the aforementioned sampling procedure. In other words, the state does not change, but the lookahead information is revealed, and the lookahead range is registered in the state.
  - If $h_t \le H$ and $B_t > 1$, we are at the middle of a batch. Then, upon playing a_t, the agent will collect a reward $R_t = \boldsymbol{R}_t(s_t, a_t, h_t)$ and transition to $s_{t+1} = \boldsymbol{s}'_t(s_t, a_t, h_t)$ with one less step remaining in the batch:

$$(s_{t+1}, \boldsymbol{R}_{t+1}, \boldsymbol{s}'_{t+1}, h_{t+1}, B_{t+1}) \leftarrow (\boldsymbol{s}'_t(s_t, a_t, h_t), \boldsymbol{R}_t, \boldsymbol{s}'_t, h_t + 1, B_t - 1).$$

    We remark that the structure of $\boldsymbol{s}'_t$ implicitly encodes the initial lookahead range (by checking what is the largest time index for which it has a nonempty element). Therefore, alongside h_t and B_t (which encodes how many steps are left in a batch), the state encodes when the batch has started and at which step of the batch the agent is found.
  - If $h_t \le H$ and $B_t = 1$, we are at the last step of a batch. Then, upon playing a_t, the agent will collect a reward $R_t = \boldsymbol{R}_t(s_t, a_t, h_t)$, transition to $s_{t+1} = \boldsymbol{s}'_t(s_t, a_t, h_t)$ and reset the lookahead information:

$$(s_{t+1}, \boldsymbol{R}_{t+1}, \boldsymbol{s}'_{t+1}, h_{t+1}, B_{t+1}) \leftarrow (\boldsymbol{s}'_t(s_t, a_t, h_t), 0^{\otimes SA\ell}, \emptyset^{\otimes SA\ell}, h_t + 1, 0),$$

    In other words, at the end of every batch, the state will always be of the form $(s, 0^{\otimes SA\ell}, \emptyset^{\otimes SA\ell}, h, 0)$.
  - If $h_t > H$, we completed H steps at the original MDP, and so w.l.o.g., we assume that the batched MDP transitions to a non-rewarding absorbing state.
  - Notice that an action B'_t only has impact when starting a new batch, and it is directly registered to the state. Therefore, with a slight abuse of notation, we use B_t to indicate both the action and the registered state component.

The horizon of the batched MDP is set to $2H$ to ensure that it passes the H^{th} step in the original MDP. This is since for every step in which $B_t = 0$ (and h_t does not change), there is at least one following state with $B_t > 0$ where h_t progresses: it holds that $h_t \ge \lceil t/2 \rceil$. In particular, $h_{2H+1} > H$ for all policies, and if $B_t > 0$ at some step in which $h_t \le H$, then $t + B_t \le 2H + 1$. Similarly, at the beginning of batches (before the lookahead range is picked), necessarily $t \le 2h - 1$.

Algorithm 3 Extended Adaptive Batching Policies

Input: Initial state s_1
Initialize: $h = 1, n = 0$ $n =$ # past batches
while $h \leq H$ **do**
  Pick $B_h = B_h(s_h, n) \in \{1, \ldots, \ell_h\}$
  Update $n \leftarrow n + 1$
  Observe B_h-step lookahead I
  for $b = 0, \ldots, B_h - 1$ **do**
    Play $a_{h+b} = \phi_{h+b}^{h,n}(s_h, I)$
    Collect reward R_{h+b} and transition to s_{h+b+1} in accordance with the lookahead I
  end for
  $h \leftarrow h + B_h$
end while

Algorithm 4 Deterministic Markov Policy in $\mathcal{M}^{\mathrm{B}}$

Input: Initial state s_1
Initialize: $h = t = 1$
while $h \leq H$ **do**
  Pick $B_t = B_t(s_h, h) \in \{1, \ldots, \ell_h\}$
  Update $t \leftarrow t + 1$
  Observe B_t-step lookahead $\boldsymbol{R}, \boldsymbol{s}'$
  for $b = 0, \ldots, B_t - 1$ **do**
    Play $a_{t+b} = \phi_{h+b}^{h,t}(s_h, \boldsymbol{R}, \boldsymbol{s}')$
    Collect reward R_{h+b} and transition to s_{h+b+1} in accordance with the lookahead $\boldsymbol{R}, \boldsymbol{s}'$
  end for
  $h \leftarrow h + B_t$, $t \leftarrow t + B_t$
end while

At any point, a policy in the batched MDP either decides on a lookahead range based on the timestep h, state s_t, and the augmented time index t, or decides on an action based on s, h, t and the lookahead information at the beginning of a batch. Thus, the only difference between such policies and ABPs in $\mathcal{M}$ is the index t. On a closer look, the only case where t advances but h does not is at the beginning of a batch: the difference $t - h$ is the number of batches started up to step h (up to 1 if we are about to start a new batch). In the case where the policy in $\mathcal{M}^{\mathrm{B}}$ is Markov and deterministic, the state s_t can be fully deduced from the state at the beginning of the batch and the lookahead information, similarly to ABPs.

In other words, a deterministic Markov policy in $\mathcal{M}^{\mathrm{B}}$ is fully equivalent to policies we term *extended adaptive batching policies*: policies that can select lookahead ranges and/or policies also depending on the number of past batches. To better illustrate the equivalence, we describe the interaction protocol of extended ABPs in $\mathcal{M}$ and deterministic Markov policies in $\mathcal{M}^{\mathrm{B}}$ in Algorithm 3 and Algorithm 4, respectively. Consequently, the values of the two MDPs coincide.

Relation to previous augmented constructions. We remark that augmenting the state space with the lookahead observations in tabular MDPs was presented in similar contexts first in one-step lookahead [16], and then also in multi-step lookahead [15, 19]. Yet, our construction directly embeds the structure of the ABPs into the environment and requires additional care due to the extra control over the lookahead range. In particular, we must account for the fact that the timesteps at which batches start can be random, and for the implicit dependence on the number of previous batches.

Values. With some abuse of notation, we use π to denote an extended batching policy in $\mathcal{M}$ and its corresponding policy in $\mathcal{M}^{\mathrm{B}}$. We let $V_t^\pi(s, \boldsymbol{R}, \boldsymbol{s}', h, B|\mathcal{M}^{\mathrm{B}})$ be the value of the policy in $\mathcal{M}^{\mathrm{B}}$. Specifically, when a batch ends, we denote the value by

$$V_t^\pi(s, 0^{\otimes SA\ell}, \emptyset^{\otimes SA\ell}, h, 0|\mathcal{M}^{\mathrm{B}}) \triangleq V_t^\pi(s, h|\mathcal{M}^{\mathrm{B}}) = \mathbb{E}_{\mathcal{M}^{\mathrm{B}}}\left[\sum_{i=t}^{2H} R_i | s_t = s, h_t = h, B_t = 0, \pi\right]$$

For an ABP, where the policy does not depend on the number of previous batches, we keep the notation from the main paper and denote the equivalent value in $\mathcal{M}$ upon starting a new batch at step h and state s by

$$V_h^\pi(s) = \mathbb{E}_{\mathcal{M}}\left[\sum_{i=h}^{H} R_i | s_h = s, E_h^n, \pi\right].$$

Recall that E_h^n is the event that the ABP π starts a new batch at timestep h, and that, by the data generation process, the expectation freshly samples all future information from the model independently between timesteps. In particular, for the described data generation process, both environments coincide at the beginning of a batch under any ABP π, and it holds that

$$V_h^\pi(s) = V_t^\pi(s, h|\mathcal{M}^{\mathrm{B}}), \qquad \forall s \in \mathcal{S}, h \in [H], t \in [2h - 1] \tag{14}$$

where the constraint to $t \in [2h-1]$ guarantees that there are enough remaining steps for the episode to end.

Finally, we use π^* to denote an optimal policy in $\mathcal{M}^{\mathrm{B}}$. Importantly, since any ABP in $\mathcal{M}$ corresponds to a policy in $\mathcal{M}^{\mathrm{B}}$, the optimal value in $\mathcal{M}^{\mathrm{B}}$ upper bounds the value of the optimal ABP; thus, by proving that the optimal policy in $\mathcal{M}^{\mathrm{B}}$ corresponds to an ABP, we will be able to derive the optimal ABP and its value. We will denote this value by $V_h^*(s)$ (and will show that it indeed does not depend on $t-h$).

Lemma 1.

1. *Let π be a deterministic Markov Policy interacting in $\mathcal{M}^{\mathrm{B}}$, as specified in Algorithm 4. Then, the value of the policy follows the Bellman equations*

$$V_t^\pi(s,h|\mathcal{M}^{\mathrm{B}}) = \mathbb{E}_{\substack{\boldsymbol{R}\sim\mathcal{R}_h^{B_t}(s),\\ \boldsymbol{s}'\sim\mathcal{P}_h^{B_t}(s)}}\left[\sum_{b=0}^{B_t-1}\mathfrak{R}_{h+b|h}(s,\phi^{h,t},\boldsymbol{R},\boldsymbol{s}') + V_{t+B_t+1}^\pi\Big(\mathfrak{s}_{h+B_t|h}(s,\phi^{h,t},\boldsymbol{R},\boldsymbol{s}'),h+B_t\Big)|\pi,s_t=s\right]$$

   for any $h \le H$, $t \le 2h-1$ and $s \in \mathcal{S}$, with the initialization $V_t^\pi(s,h|\mathcal{M}^{\mathrm{B}}) = 0$ when $h > H$ or $t > 2H$.

2. *For any $h \le H$, $t \le 2h-1$ and $s \in \mathcal{S}$, the optimal value is independent of t; we denote it by $V_h^*(s|\mathcal{M}^{\mathrm{B}})$, and it can be calculated via the optimal Bellman equations*

$$V_h^*(s|\mathcal{M}^{\mathrm{B}}) = \max_{B\in[\ell_h]}\mathbb{E}_{\substack{\boldsymbol{R}\sim\mathcal{R}_h^{B}(s),\\ \boldsymbol{s}'\sim\mathcal{P}_h^{B}(s)}}\left[\max_{\phi\in\Pi_{det}}\left\{\sum_{b=0}^{B-1}\mathfrak{R}_{h+b|h}(s,\phi,\boldsymbol{R},\boldsymbol{s}') + V_{h+B}^*\big(\mathfrak{s}_{h+B|h}(s,\phi,\boldsymbol{R},\boldsymbol{s}')\big)\right\}|s_t=s\right]$$

Proof. Before starting the proof, we again remark that the only case in which t advances in $\mathcal{M}^{\mathrm{B}}$ and h does not is a step in which a lookahead range is chosen. After each such step, a policy will be executed for at least 1 step, advancing both t and h. In particular, starting from $h=t=1$, all consecutive states must have that $t \le 2h$, and at a point before picking the lookahead range of a new batch, it holds that $t \le 2h-1$. Thus, without loss of generality, we assume throughout the proof that this always holds, which also implies that the policy necessarily plays an action at $h=H$ before the interaction ends, regardless of how it chooses the following lookahead ranges.

Recall that we denote the values at the batched MDPs of a policy π by $V_t^\pi(s,\boldsymbol{R},\boldsymbol{s}',h,B|\mathcal{M}^{\mathrm{B}})$. To ease notations, we also implicitly assume w.l.o.g. that the lookahead information $\boldsymbol{R},\boldsymbol{s}'$ is always of a valid form (can be realized in the interaction with $\mathcal{M}^{\mathrm{B}}$). We now use the dynamic programming equations for finite-horizon MDPs [21] to compute the values of π and the optimal values.

Initialization. We initialize the value to be zero for any $t > 2H$ (after the interaction ends). Moreover, by the dynamics model of $\mathcal{M}^{\mathrm{B}}$, the reward is zero starting from any state with $h > H$. Therefore, it holds that

$$V_t^\pi(s,\boldsymbol{R},\boldsymbol{s}',h,B|\mathcal{M}^{\mathrm{B}}) = 0, \qquad \forall h > H, \text{ or, } t > 2H$$

Bellman equations: arbitrary policies. Let $h \le H$. We follow the standard dynamic programming procedure [21], relying on the dynamics of $\mathcal{M}^{\mathrm{B}}$. In the middle of a batch ($B_t \ge 1$), the dynamics are deterministic (due to the already-observed lookahead information), so the equations take the form

$$V_t^\pi(s_t,\boldsymbol{R}_t,\boldsymbol{s}_t',h,B_t|\mathcal{M}^{\mathrm{B}}) = \boldsymbol{R}_t(s_t,a_t,h) + V_{t+1}^\pi(\boldsymbol{s}_t(s_t,a_t,h),\boldsymbol{R}_t,\boldsymbol{s}_t',h+1,B_t-1|\mathcal{M}^{\mathrm{B}}), \qquad B_t > 1,$$
$$V_t^\pi(s_t,\boldsymbol{R}_t,\boldsymbol{s}_t',h,B_t|\mathcal{M}^{\mathrm{B}}) = \boldsymbol{R}_t(s_t,a_t,h) + V_{t+1}^\pi(\boldsymbol{s}_t(s_t,a_t,h),0^{\otimes SA\ell},\emptyset^{\otimes SA\ell},h+1,0|\mathcal{M}^{\mathrm{B}}), \quad B_t = 1.$$

Notice that since the policy is deterministic, so is the sequence of state-actions inside the batch. Next, we apply this recursion B_t times until the end of the batch. Using $\phi^{h,t}$ to describe the deterministic policy induced by π in the middle of a batch, we get

$$\begin{aligned}V_t^\pi(s_t,\boldsymbol{R}_t,\boldsymbol{s}_t',h,B_t|\mathcal{M}^{\mathrm{B}}) &= \sum_{b=0}^{B_t-1}\boldsymbol{R}_{t+b}(s_{t+b},a_{t+b},h+b) + V_{t+B_t}^\pi(s_{t+B_t},0^{\otimes SA\ell},\emptyset^{\otimes SA\ell},h+B_t,0|\mathcal{M}^{\mathrm{B}})\\ &= \sum_{b=0}^{B_t-1}\mathfrak{R}_{h+b|h}(s_t,\phi^{h,t},\boldsymbol{R}_t,\boldsymbol{s}_t') + V_{t+B_t}^\pi(\mathfrak{s}_{h+B_t|h}(s_t,\phi^{h,t},\boldsymbol{R}_t,\boldsymbol{s}_t'),h+B_t|\mathcal{M}^{\mathrm{B}}).\end{aligned} \tag{15}$$

We now turn our focus to states where a batch is about to begin. In this case, the dynamics dictate that the state s_t remains the same without gaining a reward, B_t-step lookahead information is sampled, and the lookahead range B_t is registered in the state. Thus, the Bellman equation is

$$V_t^\pi(s,h|\mathcal{M}^{\mathrm{B}}) = V_t^\pi(s,0^{\otimes SA\ell},\emptyset^{\otimes SA\ell},h,0|\mathcal{M}^{\mathrm{B}}) = \mathbb{E}_{\substack{\boldsymbol{R}\sim\mathcal{R}_h^{B_t}(s),\\ \boldsymbol{s}'\sim\mathcal{P}_h^{B_t}(s)}}\left[V_{t+1}^\pi(s,\boldsymbol{R},\boldsymbol{s}',h,B_t|\mathcal{M}^{\mathrm{B}})\right]$$

$$= \mathbb{E}_{\substack{\boldsymbol{R}\sim\mathcal{R}_h^{B_t}(s),\\ \boldsymbol{s}'\sim\mathcal{P}_h^{B_t}(s)}}\left[\sum_{b=0}^{B_t-1}\mathfrak{R}_{h+b|h}(s,\phi^{h,t},\boldsymbol{R},\boldsymbol{s}') + V_{t+B_t+1}^\pi(\mathfrak{s}_{h+B_t|h}(s,\phi^{h,t},\boldsymbol{R},\boldsymbol{s}'),h+B_t|\mathcal{M}^{\mathrm{B}})\right],$$

where the last equality is by eq. (15). This concludes the first part of the claim.

Bellman equations: optimal value. We repeat the same process, but this time apply the optimal Bellman operator. Let $h \le H$. In the middle of a batch ($B \ge 1$), the optimal Bellman equations are

$$V_t^*(s_t,\boldsymbol{R}_t,\boldsymbol{s}_t',h,B_t|\mathcal{M}^{\mathrm{B}}) = \max_{a_t\in\mathcal{A}}\left\{\boldsymbol{R}_t(s_t,a_t,h) + V_{t+1}^*(\boldsymbol{s}_t'(s_t,a_t,h),\boldsymbol{R}_t,\boldsymbol{s}_t',h+1,B_t-1|\mathcal{M}^{\mathrm{B}})\right\}, \qquad B_t > 1,$$

$$V_t^*(s_t,\boldsymbol{R}_t,\boldsymbol{s}_t',h,B_t|\mathcal{M}^{\mathrm{B}}) = \max_{a_t\in\mathcal{A}}\left\{\boldsymbol{R}_t(s_t,a_t,h) + V_{t+1}^*(\boldsymbol{s}_t'(s_t,a_t,h),0^{\otimes SA\ell},\emptyset^{\otimes SA\ell},h+1,0|\mathcal{M}^{\mathrm{B}})\right\}, \quad B_t = 1.$$

Next, we again apply this recursion B_t times until the end of the batch, which yields

$$V_t^*(s_t,\boldsymbol{R}_t,\boldsymbol{s}_t',h,B_t|\mathcal{M}^{\mathrm{B}}) = \max_{a_t,\ldots,a_{t+B-1}\in\mathcal{A}}\left\{\sum_{b=0}^{B_t-1}\boldsymbol{R}_{t+b}(s_{t+b},a_{t+b},h+b) + V_{t+B_t}^*(s_{t+B_t},h+B_t|\mathcal{M}^{\mathrm{B}})\right\},$$

where the states implicitly depend on the sequence of actions and the lookahead information. Since the dynamics are deterministic given lookahead information, there is a full equivalence between an arbitrary sequence of actions and a deterministic Markov policy that effectively executes this sequence. Therefore, we can write the previous relation using $\phi, \mathfrak{R}$ and $\mathfrak{s}$ as

$$V_t^*(s_t,\boldsymbol{R}_t,\boldsymbol{s}_t',h,B_t|\mathcal{M}^{\mathrm{B}}) = \max_{\phi\in\Pi_{det}}\left\{\sum_{b=0}^{B_t-1}\mathfrak{R}_{h+b|h}(s_t,\phi,\boldsymbol{R}_t,\boldsymbol{s}_t') + V_{t+B_t}^*(\mathfrak{s}_{h+B_t|h}(s_t,\phi,\boldsymbol{R}_t,\boldsymbol{s}_t'),h+B_t|\mathcal{M}^{\mathrm{B}})\right\}. \tag{16}$$

Finally, as in the evaluation of fixed policies, we turn to states in which a new batch starts. Since the action a in these states is irrelevant and the valid lookahead range is $B \in [\ell_h]$, the optimal Bellman equation can be written as

$$V_t^*(s,h|\mathcal{M}^{\mathrm{B}}) = V_t^*(s,0^{\otimes SA\ell},\emptyset^{\otimes SA\ell},h,0|\mathcal{M}^{\mathrm{B}}) = \max_{B\in[\ell_h]}\mathbb{E}_{\substack{\boldsymbol{R}\sim\mathcal{R}_h^{B}(s),\\ \boldsymbol{s}'\sim\mathcal{P}_h^{B}(s)}}\left[V_{t+1}^*(s,\boldsymbol{R},\boldsymbol{s}',h,B|\mathcal{M}^{\mathrm{B}})\right]$$

$$= \max_{B\in[\ell_h]}\mathbb{E}_{\substack{\boldsymbol{R}\sim\mathcal{R}_h^{B}(s),\\ \boldsymbol{s}'\sim\mathcal{P}_h^{B}(s)}}\left[\max_{\phi\in\Pi_{det}}\left\{\sum_{b=0}^{B-1}\mathfrak{R}_{h+b|h}(s,\phi,\boldsymbol{R},\boldsymbol{s}') + V_{t+B+1}^*(\mathfrak{s}_{h+B|h}(s,\phi,\boldsymbol{R},\boldsymbol{s}'),h+B|\mathcal{M}^{\mathrm{B}})\right\}\right],$$

where the last equality substitutes eq. (16). All that is left is to prove that the optimal value only depends on h and not on t; we denote this value by $V_h^*(s|\mathcal{M}^{\mathrm{B}})$. This is the result of a simple induction. First, the claim holds trivially for $h = H+1$ and all t, where all values are zero. Now assume that the claim holds for any $h' \in \{h+1,\ldots H+1\}$, $s \in \mathcal{S}$ and $t \le 2h'-1$. In particular, for any $t \le 2h-1$ and $B \in [\ell_h]$, we have that $h+B \in \{h+1,\ldots H+1\}$ and

$$t+B+1 \le 2h+B = 2(h+B)-B \le 2(h+B)-1,$$

so by the induction assumption, $V_{t+B+1}^*(s,h+B|\mathcal{M}^{\mathrm{B}}) = V_{h+B}^*(s|\mathcal{M}^{\mathrm{B}})$. Thus, by the dynamic programming equation, we have that

$$V_t^*(s,h|\mathcal{M}^{\mathrm{B}}) = \max_{B\in[\ell_h]}\mathbb{E}_{\substack{\boldsymbol{R}\sim\mathcal{R}_h^{B}(s),\\ \boldsymbol{s}'\sim\mathcal{P}_h^{B}(s)}}\left[\max_{\phi\in\Pi_{det}}\left\{\sum_{b=0}^{B-1}\mathfrak{R}_{h+b|h}(s,\phi,\boldsymbol{R},\boldsymbol{s}') + V_{t+B+1}^*(\mathfrak{s}_{h+B|h}(s,\phi,\boldsymbol{R},\boldsymbol{s}'),h+B|\mathcal{M}^{\mathrm{B}})\right\}\right]$$

$$= \max_{B\in[\ell_h]}\mathbb{E}_{\substack{\boldsymbol{R}\sim\mathcal{R}_h^{B}(s),\\ \boldsymbol{s}'\sim\mathcal{P}_h^{B}(s)}}\left[\max_{\phi\in\Pi_{det}}\left\{\sum_{b=0}^{B-1}\mathfrak{R}_{h+b|h}(s,\phi,\boldsymbol{R},\boldsymbol{s}') + V_{h+B}^*(\mathfrak{s}_{h+B|h}(s,\phi,\boldsymbol{R},\boldsymbol{s}')|\mathcal{M}^{\mathrm{B}})\right\}\right].$$

Notably, the formula has no dependence on t and so $V_t^*(s,h|\mathcal{M}^{\mathrm{B}}) \triangleq V_h^*(s|\mathcal{M}^{\mathrm{B}})$ is independent of t. □

A direct implication of the lemma is a formula to evaluate the values of an ABP at the beginning of batches.

Corollary 2. *Let π be an ABP specified as in Algorithm 1. Then, when starting a new batch at step h and state s, the value of the policy can be calculated via the Bellman equations*

$$V_h^\pi(s) = \mathbb{E}_{I\sim\mathcal{I}_{h,B_h}(s)}\left[\sum_{b=0}^{B_h-1} \mathfrak{R}_{h+b|h}(s,\phi^h,I) + V^\pi_{h+B_h}(\mathfrak{s}_{h+B_h|h}(s,\phi^h,I))|\pi\right],$$

where $V^\pi_{H+1}(s) = 0$. Specifically, this value can be written as

$$V_h^\pi(s) = \mathbb{E}\left[\sum_{b=0}^{B_h-1} R_{h+b} + V^\pi_{h+B_h}(s_{h+B_h})|s_h = s, E_h^n, \pi\right],$$

where the expectation is over the entire episode in $\mathcal{M}$.

Proof. For ABPs, the choice of the lookahead range only depends on the state and timestep, and the policy inside a batch does not depend on the number of past batches (in contrast to extended ABPs); we denote the policy inside a batch by ϕ^h. Therefore, by Lemma 1, the value of the policy in the augmented MDP is

$$V_t^\pi(s,h|\mathcal{M}^{\mathrm{B}}) = \mathbb{E}_{\substack{\boldsymbol{R}\sim\mathcal{R}_h^{B_h}(s),\\ \boldsymbol{s}'\sim\mathcal{P}_h^{B_h}(s)}}\left[\sum_{b=0}^{B_h-1} \mathfrak{R}_{h+b|h}(s,\phi^h,\boldsymbol{R},\boldsymbol{s}') + V^\pi_{t+B_h+1}\Big(\mathfrak{s}_{h+B_h|h}(s,\phi^h,\boldsymbol{R},\boldsymbol{s}'),h+B_h\Big)|s_t=s,\pi,\mathcal{M}^{\mathrm{B}}\right],$$

initialized to zero. By the coupling between $\mathcal{M}$ and $\mathcal{M}^{\mathrm{B}}$ (that is, the data generation process in Appendix A and the related construction of $\mathcal{M}^{\mathrm{B}}$), this value coincides with the one at $\mathcal{M}$ at the beginning of any new batch

$$\begin{aligned}V_t^\pi(s,h|\mathcal{M}) &= \mathbb{E}\left[\sum_{b=0}^{B_h-1} \mathfrak{R}_{h+b|h}(s,\phi^h,I) + V^\pi_{t+B_h+1}\big(\mathfrak{s}_{h+B_h|h}(s,\phi^h,I),h+B_h\big)|s_t=s,E_h^n,\pi,\mathcal{M}\right]\\ &= \mathbb{E}_{I\sim\mathcal{I}_{h,B_h}(s)}\left[\sum_{b=0}^{B_h-1} \mathfrak{R}_{h+b|h}(s,\phi^h,I) + V^\pi_{t+B_h+1}\big(\mathfrak{s}_{h+B_h|h}(s,\phi^h,I),h+B_h\big)|\pi\right],\end{aligned}$$

where the last equality holds since the sampling model generates fresh lookahead information at the beginning of each batch. Thus, to prove the corollary, we only need to prove that the values are independent of the index t. The proof follows the exact same induction as in the proof of Lemma 1 for the optimal value, which we briefly remind. Clearly, when $h = H+1$, the initial 0-value is the same for all t. Then, assuming the claim holds by induction for all $h' > h$ and $t \le 2h'-1$, the claim trivially holds due to the dynamic programming equation at step h for all $t \le 2h-1$. Thus, by induction, for any h, if a batch starts at $t \le 2h-1$, the future expected cumulative reward in $\mathcal{M}$ is

$$V_h^\pi(s) = V_t^\pi(s,h|\mathcal{M}^{\mathrm{B}}) = \mathbb{E}_{I\sim\mathcal{I}_{h,B_h}(s)}\left[\sum_{b=0}^{B_h-1} \mathfrak{R}_{h+b|h}(s,\phi^h,I) + V^\pi_{h+B_h}\big(\mathfrak{s}_{h+B_h|h}(s,\phi^h,I)\big)|\pi\right].$$

Noticing that the batching structure enforces that $t \le 2h-1$ under all ABPs removes the restriction on t and concludes the proof of the first statement.

For the second statement, we again rely on the independent sampling of the lookahead at batch starts and write

$$\begin{aligned}V_h^\pi(s) &= \mathbb{E}_{I\sim\mathcal{I}_{h,B_h}(s)}\left[\sum_{b=0}^{B_h-1} \mathfrak{R}_{h+b|h}(s,\phi^h,I) + V^\pi_{h+B_h}(\mathfrak{s}_{h+B_h|h}(s,\phi^h,I))|\pi\right]\\ &= \mathbb{E}\left[\sum_{b=0}^{B_h-1} \mathfrak{R}_{h+b|h}(s,\phi^h,I) + V^\pi_{h+B_h}(\mathfrak{s}_{h+B_h|h}(s,\phi^h,I))|s_h=s,E_h^n,\pi\right]\\ &= \mathbb{E}\left[\sum_{b=0}^{B_h-1} R_{h+b} + V^\pi_{h+B_h}(s_{h+B_h})|s_h=s,E_h^n,\pi\right].\end{aligned}$$

The last equality is due to the definition of the lookahead information – starting from $s_h = s$, the policy ϕ^h and lookahead information I fully dictate the future collected rewards R_{h+b} and visited states s_{h+b} throughout the lookahead range. □

This corollary allows us to directly prove Proposition 3.1 by substitution.

Proposition 3.1. *There exists an optimal adaptive batching policy π^* that maximizes the value $V_1^\pi(s)$ simultaneously for all $s \in \mathcal{S}$. For any $h \in [H]$ and $s \in \mathcal{S}$, the optimal values are given by the dynamic programming equation*

$$V_h^*(s) = \max_{B \in [\ell_h]} \mathbb{E}_{I \sim \mathcal{I}_{h,B}(s)} \left[\mathfrak{Q}_h^*(s, B; I, V_{h+B}^*) \right], \tag{3}$$

where $V_{H+1}^*(s) = 0$ for all $s \in \mathcal{S}$ and the expectation samples fresh lookahead information independently between timesteps. An optimal policy chooses lookahead ranges according to

$$B_h^*(s) \in \arg\max_{B \in [\ell_h]} \mathbb{E}_{I \sim \mathcal{I}_{h,B}(s)} \left[\mathfrak{Q}_h^*(s, B; I, V_{h+B}^*) \right]. \tag{4}$$

Finally, when starting a new batch at step h and state s of length $B_h^* = B_h^*(s)$ and with lookahead information $I = I_{h,B_h^*}(s)$, it is optimal to play the deterministic policy

$$\phi^{*,h} \in \arg\max_{\phi \in \Pi_{det}} \left\{ \sum_{b=0}^{B_h^*-1} \mathfrak{R}_{h+b|h}(s, \phi, I) + V_{h+B_h^*}^* \Big(\mathfrak{s}_{h+B_h^*|h}(s, \phi, I) \Big) \right\}. \tag{5}$$

Proof. Following the coupling between the environments $\mathcal{M}$ and $\mathcal{M}^{\mathrm{B}}$, by substituting $\mathfrak{Q}_h^*(s, B; I, V_{h+B}^*)$ (see Equation (2)), $V_h^*(s)$ is equal to the optimal value in $\mathcal{M}^{\mathrm{B}}$, as stated in Lemma 1. Since $\mathcal{M}^{\mathrm{B}}$ is Markov, there exists a policy that simultaneously maximizes the value for all states and steps in the interaction. Moreover, by the construction of $\mathcal{M}^{\mathrm{B}}$, every ABP corresponds to a policy in $\mathcal{M}^{\mathrm{B}}$, so the value of ABPs is upper bounded by the optimal value in $\mathcal{M}^{\mathrm{B}}$. Thus, if we prove that an ABP achieves this value, it is necessarily an optimal ABP simultaneously for all initial states; doing so will conclude the proof. We now show that the policy π^* described in the statement indeed has a value $V_h^*(s)$.

By Corollary 2, the value of π^* follows the dynamic programming equations

$$V_h^{\pi^*}(s) = \mathbb{E}_{I \sim \mathcal{I}_{h,B_h^*}(s)} \left[\sum_{b=0}^{B_h^*-1} \mathfrak{R}_{h+b|h}(s, \phi^{*,h}, I) + V_{h+B_h^*}^{\pi^*}(\mathfrak{s}_{h+B_h^*|h}(s, \phi^{*,h}, I)) \right]. \tag{17}$$

We now prove by induction that this value is equal to the value $V_h^*(s)$. The base of the induction trivially holds when $h = H + 1$. Next, assume that the claim holds for any $h' \geq h + 1$ and $s \in \mathcal{S}$. Then, substituting the induction hypothesis in eq. (17), we get

$$\begin{aligned}
V_h^{\pi^*}(s) &= \mathbb{E}_{I \sim \mathcal{I}_{h,B_h^*}(s)} \left[\sum_{b=0}^{B_h^*-1} \mathfrak{R}_{h+b|h}(s, \phi^{*,h}, I) + V_{h+B_h^*}^{\pi^*}(\mathfrak{s}_{h+B_h^*|h}(s, \phi^{*,h}, I)) \right] \\
&= \mathbb{E}_{I \sim \mathcal{I}_{h,B_h^*}(s)} \left[\sum_{b=0}^{B_h^*-1} \mathfrak{R}_{h+b|h}(s, \phi^{*,h}, I) + V_{h+B_h^*}^{*}(\mathfrak{s}_{h+B_h^*|h}(s, \phi^{*,h}, I)) \right] \\
&\qquad \text{(by induction hypothesis)} \\
&= \mathbb{E}_{I \sim \mathcal{I}_{h,B_h^*}(s)} \left[\max_{\phi \in \Pi_{det}} \left\{ \sum_{b=0}^{B_h^*-1} \mathfrak{R}_{h+b|h}(s, \phi, I) + V_{h+B_h^*}^{*}(\mathfrak{s}_{h+B_h^*|h}(s, \phi, I)) \right\} \right] \\
&\qquad \text{(by def. of } \phi^{*,h}) \\
&= \max_{B \in [\ell_h]} \mathbb{E}_{I \sim \mathcal{I}_{h,B}(s)} \left[\max_{\phi \in \Pi_{det}} \left\{ \sum_{b=0}^{B-1} \mathfrak{R}_{h+b|h}(s, \phi, I) + V_{h+B}^{*}(\mathfrak{s}_{h+B|h}(s, \phi, I)) \right\} \right] \\
&\qquad \text{(by def. of } B_h^*) \\
&= V_h^*(s).
\end{aligned}$$

This proves the induction, showing that the value of π^* at the beginning of new batches is indeed V^*, and also concludes the proof. □

C Additional Notations

In this section, we present additional notations that we require for the algorithm and proofs.

Bonus. We first explicitly present the bonus used in our algorithm and fix it to be

$$b_h^k(s,B) = \sqrt{\frac{8\widehat{\mathrm{Var}}_h^k(s,B;\bar{V}_{h+B}^k)L_\delta^k}{n_h^k(s,B)\vee 1}} + \frac{11HL_\delta^k}{n_h^k(s,B)\vee 1}, \tag{18}$$

where $L_\delta^k = \ln\frac{18SH\ell k^3(k+1)}{\delta} \geq 1$.

Optimal Q-values and their variance. We let $Q_h^*(s,B)$ be the value of the optimal batching policy that starts a new batch of length B at time step h and state s and continues optimally. This value can be explicitly written as

$$Q_h^*(s,B) = \mathbb{E}_{I\sim\mathcal{I}_{h,B}(s)}\left[\mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)\right].$$

In particular, notice that $V_h^*(s) = \max_{B\in[\ell_h]}\{Q_h^*(s,B)\}$.

We denote by $\mathrm{Var}_h^*(s,B)$, the variance of the optimal return starting from state s at step h with lookahead range B. The variance is calculated w.r.t. fresh samples of lookahead information

$$\mathrm{Var}_h^*(s,B) = \mathbb{E}_{I\sim\mathcal{I}_{h,B}(s)}\left[\mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)^2\right] - \left(\mathbb{E}_{I\sim\mathcal{I}_{h,B}(s)}\left[\mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)\right]\right)^2. \tag{19}$$

Q-values of ABPs and their variance. For any ABP π, recall that we use the notation ϕ^h to denote the Markov policy played given a batch starting at step h and the lookahead range as $B_h = B_h^\pi(s)$. We now denote the value of π given a batch starting at step h, state s and lookahead information I to be

$$\mathfrak{V}_h^\pi(s,I) = \sum_{t=h}^{h+B_h-1} \mathfrak{R}_{t|h}(s,\phi^h,I) + V_{h+B_h}^\pi(\mathfrak{s}_{h+B_h|h}(s,\phi^h,I)). \tag{20}$$

In particular, by Corollary 2, it holds that $\mathbb{E}_{I\sim\mathcal{I}_{h,B_h}(s)}[\mathfrak{V}_h^\pi(s,I)] = V_h^\pi(s)$, and we denote the variance of this value by

$$\mathrm{Var}_h^\pi(s) = \mathbb{E}_{I\sim\mathcal{I}_{h,B_h}(s)}\left[\mathfrak{V}_h^\pi(s,I)^2\right] - \left(V_h^\pi(s)\right)^2 = \mathbb{E}_{I\sim\mathcal{I}_{h,B_h}(s)}\left[\left(\mathfrak{V}_h^\pi(s,I) - V_h^\pi(s)\right)^2\right]. \tag{21}$$

Notice that by the boundedness of the rewards (and consequently, of the values), it holds that $\mathfrak{V}_h^\pi(s,I)\in[0,H]$ and $\mathrm{Var}_h^\pi(s)\in[0,H^2]$. Moreover, since we push through the lookahead with the policy we play, assuming we start from a new batch in step h, we have that $\mathfrak{R}_{t|h}(s,\phi^h,I) = R_t$ and $\mathfrak{s}_{t|h}(s,\phi^h,I) = s_t$, so we can write

$$\mathfrak{V}_h^\pi(s,I) = \sum_{t=h}^{h+B_h-1} R_t + V_{h+B_h}^\pi(s_{h+B_h}), \text{ and}$$

$$\mathrm{Var}_h^\pi(s) = \mathbb{E}\left[\left(\sum_{t=h}^{h+B_h-1} R_t + V_{h+B_h}^\pi(s_{h+B_h}) - V_h^\pi(s)\right)^2 |\pi, s_h = s, E_h^n\right].$$

where in both expressions, the dependence in the lookahead is implicit in the rewards and states.

Finally, for ease of notation, for any $x\in\mathbb{R}^d$, define the set

$$[x,C] = \left\{y\in\mathbb{R}^d : y(i)\in[x(i),C], \forall i\in[d]\right\}.$$

D Good Events

D.1 The First Good Event – Concentration

We define the following good events:

$$E^{opt}(k) = \left\{ \forall s,h,B: \ \left| Q_h^*(s,B) - \widehat{\mathbb{E}}_{h,s,B}^k[\mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)] \right| \le \sqrt{\frac{2\mathrm{Var}_h^*(s,B)L_\delta^k}{n_h^k(s,B)\vee 1}} + \frac{HL_\delta^k}{n_h^k(s,B)\vee 1} \right\}$$

$$E^{\widehat{\mathrm{Var}}}(k) = \left\{ \forall s,h,B: \ \left| \sqrt{\widehat{\mathrm{Var}}_h^k(s,B;V_{h+B}^*)} - \sqrt{\mathrm{Var}_h^*(s,B)} \right| \le 4H\sqrt{\frac{L_\delta^k}{n_h^k(s,B)\vee 1}} \right\}$$

$$E^{Err}(k) = \Bigg\{ \forall s,h,B\in[\ell_h], \forall V\in[V_{h+B}^*,H]: \\ \widehat{\mathbb{E}}_{h,s,B}^k[\mathfrak{Q}_h^*(s,B;I,V) - \mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)] \le \left(1+\frac{1}{2H}\right)\mathbb{E}_{I\sim\mathcal{I}_{h,B}(s)}[\mathfrak{Q}_h^*(s,B;I,V) - \mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)] \\ + \frac{4H^2SL_\delta^k}{n_h^k(s,B)\vee 1} \Bigg\}$$

where we again use $L_\delta^k = \ln\frac{18SH\ell k^3(k+1)}{\delta}$. The first good event is the intersection of all these events, i.e.,

$$\mathbb{G}_1 = \bigcap_{k\ge 1} E^{opt}(k) \bigcap_{k\ge 1} E^{\widehat{\mathrm{Var}}}(k) \bigcap_{k\ge 1} E^{Err}(k),$$

and for which the following holds:

Lemma 3 (The first good event)**.** *It holds that* $\Pr(\mathbb{G}_1) \ge 1-\delta/2$.

Proof. We first remind the reader that when limiting ourselves to adaptive batching policies, upon ending a batch, the agent first decides on the next batching horizon and only then observes the lookahead information. Consequently, for such policies, the distribution of the lookahead information starting at state $s_h = s$ with lookahead range of B, $\mathcal{I}_{h,B}(s)$, is independent of any event up to timestep h (including whether $s_h = s$). Moreover, no new lookahead information is observed until the start of a new batch. As such, we can assume w.l.o.g. that the lookahead information is generated only after the agent chooses the lookahead range B, and is generated simultaneously for the whole batch of B steps.

Specifically, we assume w.l.o.g. that before the interaction starts, we generate $I_{h,B}^n(s)$ for all $h\in[H], n\in[K]$, $s\in\mathcal{S}$ and $B\in[\ell_h]$. Then, upon the i^{th} time the agent is starting a new batch of lookahead range B at $s_h = s$, the agent observes the lookahead information $I_{h,B}^i(s)$. Under this assumption, if $n_h^k(s,B) = n \ge 1$, then $\widehat{\mathbb{E}}_{h,s,B}^k[f(I)] = \frac{1}{n}\sum_{i=1}^n f(I_{h,B}^i(s))$.

As a preliminary step, we analyze the domain of the values and Q-values. First, since all rewards are non-negative, and due to the clipping of $\bar{Q}_h^k(s,B)$, it holds that $\bar{V}_h^k(s)\in[0,H-h+1]$ for all k,h and s.

Next, we recursively prove that $V_h^*(s)$, $Q_h^*(s,B)$ and $\mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)$ are all bounded (a.s.) in $[0,H-h+1]$. The recursion initializes at $V_{H+1}^*(s)=0$, obeying this requirement. We continue and assume that the domain holds by induction for $V_{h'}^*(s)$ for all $h'>h$ and $s\in\mathcal{S}$. Since the reward is bounded a.s. in $[0,1]$, then for any $s\in\mathcal{S}, B\in[\ell_h]$ and valid lookahead information I,

$$\mathfrak{Q}_h^*(s,B;I,V_{h+B}^*) = \max_{\phi\in\Pi_{det}}\left\{\sum_{t=h}^{h+B-1}\mathfrak{R}_{t|h}(s,\phi,I) + V_{h+B}^*(\mathfrak{s}_{h+B|h}(s,\phi,I))\right\} \\ \le B + (H-B-h+1) \le H-h+1,$$

and, similarly, $\mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)\ge 0$. Consequently, it holds that

$$Q_h^*(s,B) = \mathbb{E}\big[\mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)\big] \in [0,H-h+1], \quad \text{and,} \quad V_h^*(s) = \max_{B\in[\ell_h]}\{Q_h^*(s,B)\} \in [0,H-h+1].$$

Notably, the support of all values directly implies that all the events trivially hold when $n_h^k(s,B)=0$: the upper bound for this case is larger than the range of the random quantities. Thus, in the following, we analyze w.l.o.g. only the case of $n_h^k(s,B)\geq 1$.

Event $\bigcap_{k\geq 1} E^{opt}(k)$. By the assumption that the lookahead was generated independently before the interaction, it holds that $Q_h^*(s,B)=\mathbb{E}\Big[\mathfrak{Q}_h^*(s,B;I_{h,B}^i(s),V_{h+B}^*)\Big]$, and that different values of n are independent. Moreover, the variance of $\mathfrak{Q}_h^*(s,B;I_{h,B}^i(s),V_{h+B}^*)$ is $\mathrm{Var}_h^*(s,B)$. Thus, by Bernstein's inequality, for any $\delta'>0$, it holds w.p. $1-\delta'$ that

$$\begin{aligned}\left|Q_h^*(s,B)-\frac{1}{n}\sum_{i=1}^{n}\mathfrak{Q}_h^*(s,B;I_{h,B}^i(s),V_{h+B}^*)\right| &\leq \sqrt{\frac{2\mathrm{Var}_h^*(s,B)\ln\frac{2}{\delta'}}{n}}+\frac{2(H-h+1)\ln\frac{2}{\delta'}}{3n}\\ &\leq \sqrt{\frac{2\mathrm{Var}_h^*(s,B)\ln\frac{3}{\delta'}}{n}}+\frac{2H\ln\frac{3}{\delta'}}{3n} \qquad (22)\end{aligned}$$

Specifically, we get

$$\begin{aligned}&\Pr\big(\bar{E}^{opt}(k)\big)=\Pr\left\{\exists s,h,B:\left|Q_h^*(s,B)-\widehat{\mathbb{E}}_{h,s,B}^k[\mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)]\right|>\sqrt{\frac{2\mathrm{Var}_h^*(s,B)L_\delta^k}{n_h^k(s,B)\vee 1}}+\frac{HL_\delta^k}{n_h^k(s,B)\vee 1}\right\}\\
&\leq \Pr\left\{\exists s,h,B,\exists n\in[k]:\left|Q_h^*(s,B)-\widehat{\mathbb{E}}_{h,s,B}^k[\mathfrak{Q}_h^*(s,B;I,V_{h+B}^*)]\right|>\sqrt{\frac{2\mathrm{Var}_h^*(s,B)L_\delta^k}{n_h^k(s,B)\vee 1}}+\frac{HL_\delta^k}{n_h^k(s,B)\vee 1},n_h^k(s)=n\right\}\\
&=\Pr\left\{\exists s,h,B,\exists n\in[k]:\left|Q_h^*(s,B)-\frac{1}{n}\sum_{i=1}^{n}\mathfrak{Q}_h^*(s,B;I_{h,B}^i(s),V_{h+B}^*)\right|>\sqrt{\frac{2\mathrm{Var}_h^*(s,B)L_\delta^k}{n}}+\frac{HL_\delta^k}{n},n_h^k(s)=n\right\}\\
&\leq \sum_{s,h,B,n}\Pr\left\{\left|Q_h^*(s,B)-\frac{1}{n}\sum_{i=1}^{n}\mathfrak{Q}_h^*(s,B;I_{h,B}^i(s),V_{h+B}^*)\right|>\sqrt{\frac{2\mathrm{Var}_h^*(s,B)L_\delta^k}{n}}+\frac{HL_\delta^k}{n}\right\}\\
&\overset{(*)}{\leq} SH\ell k\frac{\delta}{6SH\ell k^3(k+1)}\leq\frac{\delta}{6k(k+1)},\end{aligned}$$

where $(*)$ is by eq. (22), since L_δ^k corresponds to the value $\delta'=\frac{\delta}{6SH\ell k^3(k+1)}$. Using the union bound to sum over all $k\geq 1$ yields

$$\Pr\left\{\bigcap_{k\geq 1}E^{opt}(k)\right\}=1-\Pr\left\{\bigcup_{k\geq 1}\bar{E}^{opt}(k)\right\}\geq 1-\sum_{k\geq 1}\Pr\big\{\bar{E}^{opt}(k)\big\}\geq 1-\sum_{k\geq 1}\frac{\delta}{6k(k+1)}\geq 1-\frac{\delta}{6}.$$

Event $\bigcap_{k\geq 1} E^{\widehat{\mathrm{Var}}}(k)$. Again, by the independent initial lookahead generation, when $n_h^k(s,B)=n$, the variance $\widehat{\mathrm{Var}}_h^k(s,B;V_{h+B}^*)$ is calculated using the n i.i.d. samples $\Big\{I_{h,B}^i(s)\Big\}_{i\in[n]}$. We now apply Lemma 10 on the variables $X_i=\mathfrak{Q}_h^*(s,B;I_{h,B}^i(s),V_{h+B}^*)$ (which are bounded in $[0,H]$), and as with $\bar{E}^{opt}(k)$, we take the union bound on $n\in[k]$ to account for all possible values of $n_h^k(s,B)$. Also taking the union bounds for all potential s,h,B, and noting that L_δ^k corresponds to setting $\delta'=\frac{\delta}{9SH\ell k^3(k+1)}$, we get

$$\begin{aligned}\Pr\Big(\bar{E}^{\widehat{\mathrm{Var}}}(k)\Big)&=\Pr\left\{\exists s,h,B:\left|\sqrt{\widehat{\mathrm{Var}}_h^k(s,B;V_{h+B}^*)}-\sqrt{\mathrm{Var}_h^*(s,B)}\right|>4H\sqrt{\frac{L_\delta^k}{n_h^k(s,B)\vee 1}}\right\}\\
&\leq SH\ell k\frac{\delta}{9SH\ell k^3(k+1)}\leq\frac{\delta}{6k(k+1)}.\end{aligned}$$

Therefore, $\Pr\Big(\bigcap_{k\geq 1}E^{\widehat{\mathrm{Var}}}(k)\Big)\geq 1-\sum_{k\geq 1}\Pr\Big(\bar{E}^{\widehat{\mathrm{Var}}}(k)\Big)\geq 1-\frac{\delta}{6}$.

Event $\bigcap_{k\geq 1} E^{Err}(k)$. For this event, we apply Lemma 8 for any s,h,B w.r.t. the random vectors $X_i(s')=\mathfrak{J}_{h,B}^{I_{h,B}^i(s)}(s,s')$ and $Y_i=\mathfrak{S}_{h,B}^{I_{h,B}^i(s)}(s)$; by the boundedness of the rewards $X_i\in[0,H]^S$,

and clearly $Y_i \subseteq [S]$. Moreover, for this choice, we get by eq. (8) that

$$f(X_i, Y_i, V) = \max_{s' \in Y_i} \{X_i(s') + V(s')\} = \mathfrak{Q}^*_h(s, B; I^i_{h,B}(s), V).$$

In particular, fixing $V_0 = V^*_{h+B}$, $\delta' = \frac{\delta}{6SH\ell k^2(k+1)}$, $\alpha = 1/2H$ and using the union bound similarly to the events $E^{opt}(k), E^{\widehat{\mathrm{Var}}}(k)$, Lemma 8 implies that $\Pr\{\bar{E}^{Err}(k)\} \leq \frac{\delta}{6k(k+1)}$. Finally, another union bound over all $k \geq 1$ leads to the bound $\Pr\left(\bigcap_{k\geq 1} E^{Err}(k)\right) \geq 1 - \frac{\delta}{6}$.

To conclude, we proved that each of the events $\bigcap_{k\geq 1} E^{opt}(k) \bigcap_{k\geq 1} E^{\widehat{\mathrm{Var}}}(k) \bigcap_{k\geq 1} E^{Err}(k)$ hold w.p. at least $1 - \frac{\delta}{6}$; thus, as their intersection, the event $\mathbb{G}_1$ holds w.p. at least $1 - \frac{\delta}{2}$. □

D.2 Optimism Under the First Good Event

We now prove that under the event $\mathbb{G}_1$, the values that AL-UCB outputs are optimistic.

Lemma 4 (Optimism)**.** *Under the first good event $\mathbb{G}_1$, for all $k \in [K]$, $h \in [H]$, $s \in \mathcal{S}$, $B \in [\ell_h]$ and lookahead information I, it holds that $\mathfrak{Q}^*_h(s, B; I, V^*_{h+B}) \leq \mathfrak{Q}^*_h(s, B; I, \bar{V}^k_{h+B})$; in particular, $V^*_h(s) \leq \bar{V}^k_h(s)$.*

Proof. We prove this by backwards induction on the values $V^*_h(s)$ and $\bar{V}^k_h(s)$. First, by definition, $V^*_{H+1}(s) = \bar{V}^k_{H+1}(s) = 0$ for all $s \in \mathcal{S}$, and so the base of the induction holds. Now, for any $k \in [K]$ and $h \in [H]$, assume by induction that $V^*_t(s) \leq \bar{V}^k_t(s)$ for all $t \in \{h+1, \ldots, H+1\}$ and $s \in \mathcal{S}$; we will show that it implies both claims also for timestep h.

For the first claim, by the induction hypothesis, for any $s \in \mathcal{S}$, $B \in [\ell_h]$ and lookahead information I, it holds that

$$\begin{aligned}
\mathfrak{Q}^*_h(s, B; I, \bar{V}^k_{h+B}) &= \max_{s' \in \mathfrak{S}^I_{h,B}(s)} \left\{ \mathfrak{J}^I_{h,B}(s, s') + \bar{V}^k_{h+B}(s') \right\} \\
&\geq \max_{s' \in \mathfrak{S}^I_{h,B}(s)} \left\{ \mathfrak{J}^I_{h,B}(s, s') + V^*_{h+B}(s') \right\} \quad (\bar{V}^k_{h+B}(s') \geq V^*_{h+B}(s') \text{ since } B \geq 1) \\
&= \mathfrak{Q}^*_h(s, B; I, V^*_{h+B}). \qquad (23)
\end{aligned}$$

We now continue by proving that under the good event and induction hypothesis, $Q^*_h(s, B) \leq \bar{Q}^k_h(s, B)$ for all $s \in \mathcal{S}$ and $B \in [\ell_h]$. We first briefly discuss the domain of the Q-values. Since rewards are bounded in $[0, 1]$, it holds that $Q^*_h(s, B) \in [0, H - h + 1]$. As for the optimistic values, the rewards are non-negative, and therefore so are $\bar{V}^k_h(s)$ and $\mathfrak{Q}^*_h(s, B; I, \bar{V}^k_{h+B})$ (almost surely for all potential lookahead realizations). Moreover, again by the upper bound on the rewards and the truncation of $\bar{V}^k_{h+B}$, it holds that $\mathfrak{Q}^*_h(s, B; I, \bar{V}^k_{h+B}) \leq H - h + 1$ almost surely. Thus, to summarize, we know that both $Q^*_h(s, B)$ and $\mathfrak{Q}^*_h(s, B; I, \bar{V}^k_{h+B})$ take values in $[0, H - h + 1]$. In particular, whenever $\bar{Q}^k_h(s, B) = H - h + 1$, the claim $Q^*_h(s, B) \leq \bar{Q}^k_h(s, B)$ trivially holds; therefore, in the following, we assume w.l.o.g. that $\bar{Q}^k_h(s, B) < H - h + 1$. Notice that this also implies that $n^k_h(s, B) \geq 1$: when $n^k_h(s, B) = 0$, the bonus is large enough to ensure that $\bar{Q}^k_h(s, B) = H - h + 1$.

Using the aforementioned eq. (23) (and the insight that $n^k_h(s, B) \geq 1$) to apply Corollary 14 with $\alpha = \sqrt{\frac{8L^k_\delta}{n^k_h(s,B) \vee 1}}$ and a value upper bound of H:

$$\begin{aligned}
\bar{Q}^k_h(s, B) &= \widehat{\mathbb{E}}^k_{h,s,B}\left[\mathfrak{Q}^*_h(s, B; I, \bar{V}^k_{h+B})\right] + \sqrt{\frac{8\widehat{\mathrm{Var}}^k_h(s, B; \bar{V}^k_{h+B})L^k_\delta}{n^k_h(s, B) \vee 1}} + \frac{11HL^k_\delta}{n^k_h(s, B) \vee 1} \\
&\geq \left(\widehat{\mathbb{E}}^k_{h,s,B}\left[\mathfrak{Q}^*_h(s, B; I, \bar{V}^k_{h+B})\right] + \max\left\{ \sqrt{\frac{8\widehat{\mathrm{Var}}^k_h(s, B; \bar{V}^k_{h+B})L^k_\delta}{n^k_h(s, B) \vee 1}}, \frac{8HL^k_\delta}{n^k_h(s, B) \vee 1} \right\} \right) + \frac{3HL^k_\delta}{n^k_h(s, B) \vee 1} \\
&\geq \left(\widehat{\mathbb{E}}^k_{h,s,B}[\mathfrak{Q}^*_h(s, B; I, V^*_{h+B})] + \max\left\{ \sqrt{\frac{8\widehat{\mathrm{Var}}^k_h(s, B; V^*_{h+B})L^k_\delta}{n^k_h(s, B) \vee 1}}, \frac{8HL^k_\delta}{n^k_h(s, B) \vee 1} \right\} \right) + \frac{3HL^k_\delta}{n^k_h(s, B) \vee 1} \\
&\qquad \text{(Corollary 14)} \\
&\geq \left(\widehat{\mathbb{E}}^k_{h,s,B}[\mathfrak{Q}^*_h(s, B; I, V^*_{h+B})] + \sqrt{\frac{2\widehat{\mathrm{Var}}^k_h(s, B; V^*_{h+B})L^k_\delta}{n^k_h(s, B) \vee 1}} + \frac{4HL^k_\delta}{n^k_h(s, B) \vee 1} \right) + \frac{3HL^k_\delta}{n^k_h(s, B) \vee 1} \\
&\qquad (\max\{a, b\} \geq \tfrac{a+b}{2}) \\
&\geq \widehat{\mathbb{E}}^k_{h,s,B}[\mathfrak{Q}^*_h(s, B; I, V^*_{h+B})] + \sqrt{\frac{2\mathrm{Var}^*_h(s, B)L^k_\delta}{n^k_h(s, B) \vee 1}} + \frac{HL^k_\delta}{n^k_h(s, B) \vee 1} \qquad \text{(Under } E^{\widehat{\mathrm{Var}}}\text{)} \\
&\geq Q^*_h(s, B). \qquad \text{(Under } E^{opt}\text{)}
\end{aligned}$$

This directly implies that

$$\bar{V}^k_h(s) = \max_{B \in [\ell_h]} \left\{ \bar{Q}^k_h(s, B) \right\} \geq \max_{B \in [\ell_h]} \{Q^*_h(s, B)\} = V^*_h(s),$$

completing the proof of the induction. □

D.3 The Second Good Event – Cumulative Concentration

Define the following events:

$$E^{\mathrm{Var}} = \left\{ K \geq 1 : \sum_{k=1}^{K}\sum_{i=1}^{H} \mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k) \leq 2\sum_{k=1}^{K}\sum_{i=1}^{H} \mathbb{E}\Big[\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)|s_1^k, \pi^k\Big] + 4H^3 \ln\frac{4K(K+1)}{\delta} \right\},$$

$$E^{\mathrm{diff}} = \Bigg\{ \forall i \in [H], K \geq 1 : \sum_{k=1}^{K} \mathbb{E}\Big[\bar{V}_{\mathfrak{t}_{i+1}^k}^k(s_{\mathfrak{t}_{i+1}^k}^k) - V_{\mathfrak{t}_{i+1}^k}^{\pi^k}(s_{\mathfrak{t}_{i+1}^k}^k)|\mathfrak{t}_i^k, s_{\mathfrak{t}_i^k}^k, \pi^k\Big]$$
$$\leq \left(1+\frac{1}{2H}\right)\sum_{k=1}^{K}\Big(\bar{V}_{\mathfrak{t}_{i+1}^k}^k(s_{\mathfrak{t}_{i+1}^k}^k) - V_{\mathfrak{t}_{i+1}^k}^{\pi^k}(s_{\mathfrak{t}_{i+1}^k}^k)\Big) + 18H^2\ln\frac{4HK(K+1)}{\delta} \Bigg\},$$

where $\mathrm{Var}_h^\pi(s)$ is defined at Equation (21).

The second good event is the intersection of the events $\mathbb{G}_2 = E^{\mathrm{diff}} \cap E^{\mathrm{Var}}$. We define the good event $\mathbb{G} = \mathbb{G}_1 \cap \mathbb{G}_2$.

Lemma 5. *The good event $\mathbb{G}$ holds with a probability of at least $1-\delta$.*

Proof. **Event E^{Var}.** Let $\mathcal{F}_k$ be the natural filtration including all information on the interaction of the algorithm up to the end of episode k (states, actions, batch starting times, lookahead ranges/information), as well as the initial state s_1^{k+1}. In particular, π^k is $\mathcal{F}_{k-1}$-measurable and $\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)$ is $\mathcal{F}_k$-measurable. Moreover, since rewards are bounded a.s. in $[0,1]$, $\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k) \in [0,H^2]$ and $\sum_{i=1}^H \mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)$ is a.s. in $[0,H^3]$. Then, by Lemma 15, for any $K \geq 1$, it holds w.p. $1-\frac{\delta}{4K(K+1)}$ that

$$\sum_{k=1}^{K}\sum_{i=1}^{H} \mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k) \leq 2\sum_{k=1}^{K}\sum_{i=1}^{H} \mathbb{E}\Big[\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)|\mathcal{F}_{k-1}\Big] + 4H^3\ln\frac{4K(K+1)}{\delta}$$

Taking the union bound for all $K \geq 1$ (and recalling that $\sum_{n\geq 1}\frac{1}{n(n+1)} = 1$), the aforementioned event holds uniformly for all $K \geq 1$ w.p. $1-\delta/4$. Finally, we note that given π^k and s_1^k, the k^{th} episode is independent of all events before its start, and since both π^k and s_1^k are $\mathcal{F}_{k-1}$-measurable,

$$\mathbb{E}\Big[\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)|\mathcal{F}_{k-1}\Big] = \mathbb{E}\Big[\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)|s_1^k, \pi^k\Big].$$

Substituting this, we get that $\Pr\big(E^{\mathrm{Var}}\big) \geq 1-\delta/4$.

Event E^{diff}. Let $i \in [H]$ and let $\mathcal{F}_{k,i}$ be the filtration that includes all information in F_{k-1}, as well as all information on the interaction in episode k up to the beginning of the i^{th} lookahead batch, including its starting time $\mathfrak{t}_i^k$ and state $s_{\mathfrak{t}_i^k}^k$. Specifically, π^k, $\bar{V}_h^k$ and $V_h^{\pi^k}$ are all $\mathcal{F}_{k,i}$-measurable and the value difference $\bar{V}_{\mathfrak{t}_{i+1}^k}^k(s_{\mathfrak{t}_{i+1}^k}^k) - V_{\mathfrak{t}_{i+1}^k}^{\pi^k}(s_{\mathfrak{t}_{i+1}^k}^k)$ is $\mathcal{F}_{k+1,i}$-measurable. Moreover, the policy π^k decides on the lookahead range only depending on the timestep and state, so we can directly calculate the beginning of the next batch given the starting point of the current one: $\mathfrak{t}_{i+1}^k$ is $\mathcal{F}_{k,i}$-measurable. Consequently, given $\mathcal{F}_{k,i}$, the only randomness in $\bar{V}_{\mathfrak{t}_{i+1}^k}^k(s_{\mathfrak{t}_{i+1}^k}^k) - V_{\mathfrak{t}_{i+1}^k}^{\pi^k}(s_{\mathfrak{t}_{i+1}^k}^k)$ is due to $s_{\mathfrak{t}_{i+1}^k}^k$, which in turn only depends on the lookahead generated inside the batch and the policy. This lookahead information only depends on the starting point of the batch, and thus

$$\mathbb{E}\Big[\bar{V}_{\mathfrak{t}_{i+1}^k}^k(s_{\mathfrak{t}_{i+1}^k}^k) - V_{\mathfrak{t}_{i+1}^k}^{\pi^k}(s_{\mathfrak{t}_{i+1}^k}^k)|\mathcal{F}_{k,i}\Big] = \mathbb{E}\Big[\bar{V}_{\mathfrak{t}_{i+1}^k}^k(s_{\mathfrak{t}_{i+1}^k}^k) - V_{\mathfrak{t}_{i+1}^k}^{\pi^k}(s_{\mathfrak{t}_{i+1}^k}^k)|\mathfrak{t}_i^k, s_{\mathfrak{t}_i^k}^k, \pi^k\Big]. \tag{24}$$

Finally, define

$$W_k = \mathbb{1}\Big\{\forall s \in \mathcal{S}, h \in [H] : \bar{V}_h^k(s) - V_h^{\pi^k}(s) \geq 0\Big\}, \qquad \text{and,} \qquad Y_{k,i} = W_k\Big(\bar{V}_{\mathfrak{t}_{i+1}^k}^k(s_{\mathfrak{t}_{i+1}^k}^k) - V_{\mathfrak{t}_{i+1}^k}^{\pi^k}(s_{\mathfrak{t}_{i+1}^k}^k)\Big).$$

Notice that W_k is $F_{k,i}$-measurable for all i. Furthermore, by Lemma 4, when $\mathbb{G}_1$ holds, then $\bar{V}_h^k(s) \geq V_h^*(s) \geq V_h^{\pi^k}(s)$, and we have that $W_k = 1$.

By the clipping of $\bar{Q}_h^k$, it holds that $Y_{k,i} \in [0, H]$ (and the same trivially holds in cases where $\mathfrak{t}_{i+1}^k = H+1$: in this case, both values are zero). We can thus apply Lemma 15 and get that for any $K \geq 1$ and $i \in [H]$, w.p. $1 - \frac{\delta}{4HK(K+1)}$,

$$\sum_{k=1}^{K} \mathbb{E}[Y_{k,i} | \mathcal{F}_{k,i}] \leq \left(1 + \frac{1}{2H}\right) \sum_{k=1}^{K} Y_{k,i} + 18H^2 \ln \frac{4HK(K+1)}{\delta}.$$

Taking the union bound, the same holds uniformly for all $K \geq 1$ and $i \in [H]$ w.p. at least $1 - \delta/4$. We now relate this to the event E^{diff}, when $\mathbb{G}_1$ holds.

$$\begin{aligned}
\mathbb{G}_1 \cap E^{\text{diff}} &= \mathbb{G}_1 \cap \Bigg\{ \forall i \in [H], K \geq 1 : \sum_{k=1}^{K} \mathbb{E}\Big[\bar{V}^k_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) - V^{\pi^k}_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) | \mathfrak{t}^k_i, s^k_{\mathfrak{t}^k_i}, \pi^k \Big] \\
&\qquad \leq \left(1 + \frac{1}{2H}\right) \sum_{k=1}^{K} \Big(\bar{V}^k_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) - V^{\pi^k}_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) \Big) + 18H^2 \ln \frac{4HK(K+1)}{\delta} \Bigg\} \\
&= \mathbb{G}_1 \cap \Bigg\{ \forall i \in [H], K \geq 1 : \sum_{k=1}^{K} \mathbb{E}\Big[\bar{V}^k_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) - V^{\pi^k}_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) | \mathcal{F}_{k,i} \Big] \\
&\qquad \leq \left(1 + \frac{1}{2H}\right) \sum_{k=1}^{K} \Big(\bar{V}^k_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) - V^{\pi^k}_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) \Big) + 18H^2 \ln \frac{4HK(K+1)}{\delta} \Bigg\} \\
&\qquad\qquad \text{(by eq. (24))} \\
&\overset{(*)}{=} \mathbb{G}_1 \cap \Bigg\{ \forall i \in [H], K \geq 1 : \sum_{k=1}^{K} \mathbb{E}\Big[W_k \Big(\bar{V}^k_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) - V^{\pi^k}_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) \Big) | \mathcal{F}_{k,i} \Big] \\
&\qquad \leq \left(1 + \frac{1}{2H}\right) \sum_{k=1}^{K} W_k \Big(\bar{V}^k_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) - V^{\pi^k}_{\mathfrak{t}^k_{i+1}}(s^k_{\mathfrak{t}^k_{i+1}}) \Big) + 18H^2 \ln \frac{4HK(K+1)}{\delta} \Bigg\} \\
&= \mathbb{G}_1 \cap \underbrace{\left\{ \forall i \in [H], K \geq 1 : \sum_{k=1}^{K} \mathbb{E}[Y_{k,i} | \mathcal{F}_{k,i}] \leq \left(1 + \frac{1}{2H}\right) \sum_{k=1}^{K} Y_{k,i} + 18H^2 \ln \frac{4HK(K+1)}{\delta} \right\}}_{\tilde{E}^{\text{diff}}},
\end{aligned}$$

where relation $(*)$ holds since W_k is $\mathcal{F}_{k,i}$-measurable and equals 1 under $\mathbb{G}_1$. In particular, we know that $\mathbb{G}_1$ holds w.p. $1 - \delta/2$ (by Lemma 3) and $\tilde{E}^{\text{diff}}$ holds w.p. $1 - \delta/4$, and so $\Pr\Big\{\overline{\mathbb{G}_1 \cap E^{\text{diff}}}\Big\} \leq 3\delta/4$.

Event $\mathbb{G}$. Combining both parts, we have

$$\Pr\{\overline{\mathbb{G}}\} = \Pr\Big\{\overline{\mathbb{G}_1 \cap E^{\text{diff}} \cap E^{\text{Var}}}\Big\} \leq \Pr\Big\{\overline{E^{\text{Var}}}\Big\} + \Pr\Big\{\overline{\mathbb{G}_1 \cap E^{\text{diff}}}\Big\} \leq \delta/4 + 3\delta/4 = \delta.$$

□

E Regret Analysis

Theorem 4.1. *For any $\delta \in (0,1)$, w.p. at least $1-\delta$ the regret of AL-UCB is upper bounded by* $\mathrm{Reg}(K) = \mathcal{O}\Big(\sqrt{H^3SK\ell}\ln\frac{SH\ell K}{\delta} + H^3S^2\ell\ln^2\frac{SH\ell K}{\delta}\Big)$.

Proof. Throughout the proof, we assume that the good event $\mathbb{G}$ holds; therefore, by Lemma 5, the resulting bounds will hold w.p. at least $1-\delta$. Fix $k \in [K], h \in [H]$ and $s \in \mathcal{S}$. We start by rewriting the value $V_h^{\pi^k}(s)$ using the notation $\mathfrak{Q}^*$. To ease notations, we now shorten $B_h^{\pi^k}(s) \to B_h$ and $\mathbb{E}_{I\sim\mathcal{I}_{h,B_h}(s)}[\cdot] \to \mathbb{E}_I[\cdot]$.

$$
\begin{aligned}
V_h^{\pi^k}(s) &= \mathbb{E}_I\left[\sum_{t=h}^{h+B_h-1} \mathfrak{R}_{t|h}(s,\phi_h^k,I) + V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I))\right] && \text{(By Corollary 2)}\\
&= \mathbb{E}_I\left[\sum_{t=h}^{h+B_h-1} \mathfrak{R}_{t|h}(s,\phi_h^k,I) + \bar{V}_{h+B_h}^{k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I))\right]\\
&\quad + \mathbb{E}_I\Big[V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I)) - \bar{V}_{h+B_h}^{k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I))\Big]\\
&= \mathbb{E}_I\left[\max_{\phi\in\Pi_{det}}\left\{\sum_{t=h}^{h+B_h-1} \mathfrak{R}_{t|h}(s,\phi,I) + \bar{V}_{h+B_h}^{k}(\mathfrak{s}_{h+B_h|h}(s,\phi,I))\right\}\right]\\
&\quad + \mathbb{E}_I\Big[V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I)) - \bar{V}_{h+B_h}^{k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I))\Big]\\
&\qquad\qquad \text{(Def. of } \phi_h^k \text{ in Equation (12))}\\
&= \mathbb{E}_I\big[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k)\big] + \mathbb{E}_I\Big[V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I)) - \bar{V}_{h+B_h}^{k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I))\Big].
\end{aligned}
$$

In particular, by optimism (Lemma 4), we have $V_h^*(s) - V_h^{\pi^k}(s) \le \bar{V}_h^k(s) - V_h^{\pi^k}(s)$: we henceforth bound the difference between the optimistic and real value of π^k.

$$
\begin{aligned}
\bar{V}_h^k(s) - V_h^{\pi^k}(s) &= \bar{Q}_h^k(s,B_h) - V_h^{\pi^k}(s)\\
&\le \widehat{\mathbb{E}}_{h,s,B_h}^k\Big[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k)\Big] + b_h^k(s,B_h) - \mathbb{E}_I\Big[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k)\Big]\\
&\quad + \mathbb{E}_I\Big[\bar{V}_{h+B_h}^{k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I)) - V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I))\Big]\\
&\le \underbrace{\widehat{\mathbb{E}}_{h,s,B_h}^k\Big[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\Big] - \mathbb{E}_I\Big[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\Big]}_{(i)}\\
&\quad + \underbrace{\widehat{\mathbb{E}}_{h,s,B_h}^k\big[\mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\big] - \mathbb{E}_I\big[\mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\big]}_{(ii)} + b_h^k(s,B_h)\\
&\quad + \mathbb{E}_I\Big[\bar{V}_{h+B_h}^{k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I)) - V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi_h^k,I))\Big] \qquad (25)
\end{aligned}
$$

Before continuing, we take a brief detour to focus on bounding the bonus.

Bounding the bonus. To bound $b_h^k(s,B_h)$, we have to bound the empirical variance. In the following, we aim to relate it to the variance of the value of π^k. We start with some additional notations. Recall that in Appendix C, we denote the value of an ABP π given lookahead I by $\mathfrak{V}_h^\pi(s,I)$ and its variance w.r.t. the lookahead as $\mathrm{Var}_h^\pi(s)$. Since both values and rewards are non-negative, it holds that $\mathfrak{V}_h^\pi(s,I) \ge 0$. Moreover, by the optimality of V^*,

$$
\begin{aligned}
\mathfrak{V}_h^\pi(s,I) &\le \sum_{t=h}^{h+B_h^\pi-1} \mathfrak{R}_{t|h}(s,\phi^h,I) + V_{h+B_h^\pi}^*(\mathfrak{s}_{h+B_h^\pi|h}(s,\phi^h,I))\\
&\le \max_\phi\left\{\sum_{t=h}^{h+B_h^\pi-1} \mathfrak{R}_{t|h}(s,\phi,I) + V_{h+B_h^\pi}^*(\mathfrak{s}_{h+B_h^\pi|h}(s,\phi,I))\right\}\\
&= \mathfrak{Q}_h^*(s,B_h^\pi;I,V_{h+B_h^\pi}^*),
\end{aligned}
$$

where the inequality holds for all lookahead information I. going back to π^k and again abbreviate $B_h = B_h^{\pi^k}(s)$, by Lemma 4 and under the good event,

$$0 \le \underbrace{\mathfrak{V}_h^{\pi^k}(s,I)}_{'V_1'} \le \underbrace{\mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)}_{'V_2'} \le \underbrace{\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k)}_{'V_3'} \le H. \tag{26}$$

Therefore, we can apply Lemma 12 w.r.t. both the empirical and real lookahead probability measures (that is, generate I with the measures corresponding to $\widehat{\mathbb{E}}_{h,s,B}^k$ and $\mathbb{E}_I$). Specifically, we set $\alpha = 4\sqrt{8L_\delta^k}H$, $n = n_h^k(s,B_h) \vee 1$ and $\beta = 4H\sqrt{\frac{L_\delta^k}{n_h^k(s,B_h)\vee 1}}$, so that the variance difference assumption holds under the event $E^{\widehat{\mathrm{Var}}}(k)$. For these values, Lemma 12 claims that

$$\begin{aligned}\frac{\sqrt{8\widehat{\mathrm{Var}}_h^k(s,B_h;\bar{V}_{h+B_h}^k)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} &\le \frac{\sqrt{8\mathrm{Var}_h^{\pi^k}(s)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} + \frac{1}{4H}\widehat{\mathbb{E}}_{h,s,B_h}^k\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\right] \\ &\quad + \frac{1}{4H}\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*) - \mathfrak{V}_h^{\pi^k}(s,I)\right] + \underbrace{\frac{16H^2L_\delta^k}{n_h^k(s,B_h)\vee 1}}_{\sqrt{8L_\delta^k}C\alpha/(2n)} + \underbrace{\frac{12H^2L_\delta^k}{n_h^k(s,B_h)\vee 1}}_{\sqrt{8L_\delta^k}\beta/\sqrt{n}}.\end{aligned}$$

We remark that when $n_h^k(s,B_h) = 0$, the empirical variance is zero and the above inequality holds trivially. Next, we further rely on the good event, and specifically, the event $E^{Err}(k)$; by the optimism of $\bar{V}^k$, we can bound the empirical expectation by the real expectation and get the bound

$$\begin{aligned}\frac{\sqrt{8\widehat{\mathrm{Var}}_h^k(s,B_h;\bar{V}_{h+B_h}^k)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} &\le \frac{\sqrt{8\mathrm{Var}_h^{\pi^k}(s)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} + \frac{1}{4H}\left(1+\frac{1}{2H}\right)\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\right] + \frac{HSL_\delta^k}{n_h^k(s,B_h)\vee 1} \\ &\quad + \frac{1}{4H}\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*) - \mathfrak{V}_h^{\pi^k}(s,I)\right] + \frac{28H^2L_\delta^k}{n_h^k(s,B_h)\vee 1} \\ &\overset{(*)}{\le} \frac{\sqrt{8\mathrm{Var}_h^{\pi^k}(s)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} + \frac{1}{4H}\left(1+\frac{1}{2H}\right)\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{V}_h^{\pi^k}(s,I)\right] \\ &\quad + \frac{1}{4H}\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{V}_h^{\pi^k}(s,I)\right] + \frac{29H^2SL_\delta^k}{n_h^k(s,B_h)\vee 1} \\ &\le \frac{\sqrt{8\mathrm{Var}_h^{\pi^k}(s)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} + \frac{1}{2H}\left(1+\frac{1}{2H}\right)\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{V}_h^{\pi^k}(s,I)\right] + \frac{29H^2SL_\delta^k}{n_h^k(s,B_h)\vee 1}\end{aligned}$$

In inequality $(*)$, we relied on eq. (26) to lower/upper bound the values so that the argument of both expectations will be the same, and combined the $1/n$ terms. In the last inequality, we also slightly increased the coefficient of the expectation – this is valid since optimism ensures that the expectation is positive.

Finally, we bound the value difference $\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{V}_h^{\pi^k}(s,I)$. Let $\phi^{k,h}$ be the Markov policy played by π^k inside a batch starting at step h and state s given lookahead information I. Then,

$$\begin{aligned}&\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{V}_h^{\pi^k}(s,I) \\ &\quad= \left(\max_{\phi\in\Pi_{det}}\left\{\sum_{t=h}^{h+B_h-1}\mathfrak{R}_{t|h}(s,\phi,I) + \bar{V}_{h+B_h}^k(\mathfrak{s}_{h+B_h|h}(s,\phi,I))\right\}\right) \\ &\quad- \left(\sum_{t=h}^{h+B_h-1}\mathfrak{R}_{t|h}(s,\phi^{k,h},I) + V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I))\right) \\ &\quad= \left(\sum_{t=h}^{h+B_h-1}\mathfrak{R}_{t|h}(s,\phi^{k,h},I) + \bar{V}_{h+B_h}^k(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I))\right) \\ &\quad- \left(\sum_{t=h}^{h+B_h-1}\mathfrak{R}_{t|h}(s,\phi^{k,h},I) + V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I))\right) \qquad \text{(Def. of } \phi^{k,h}\text{)} \\ &\quad= \bar{V}_{h+B_h}^k(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I)) - V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I)).\end{aligned} \tag{27}$$

Plugging this into the variance bound and then into the bonus leads to the desired bound on the bonus:

$$b_h^k(s,B_h) = \sqrt{\frac{8\widehat{\mathrm{Var}}_h^k(s,B_h;\bar{V}_{h+B_h}^k)L_\delta^k}{n_h^k(s,B)\vee 1}} + \frac{11HL_\delta^k}{n_h^k(s,B)\vee 1}$$
$$\leq \frac{\sqrt{8\mathrm{Var}_h^{\pi^k}(s)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} + \frac{1}{2H}\left(1+\frac{1}{2H}\right)\mathbb{E}_I\left[\bar{V}_{h+B_h}^k(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I)) - V_{h+B_h}^{\pi}(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I))\right] + \frac{40H^2SL_\delta^k}{n_h^k(s,B_h)\vee 1}.$$

Bounding term (i)**.** Next, we bound term (i), relying on similar concepts as the bonus bound. We again utilize the optimism and event $E^{Err}(k)$ to bound

$$(i) = \widehat{\mathbb{E}}_{h,s,B_h}^k\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\right] - \mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\right]$$
$$\leq \frac{1}{2H}\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\right] + \frac{4H^2SL_\delta^k}{n_h^k(s,B_h)\vee 1}$$
(Under $E^{Err}(k)$+optimism)
$$\leq \frac{1}{2H}\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{V}_h^{\pi^k}(s,I)\right] + \frac{4H^2SL_\delta^k}{n_h^k(s,B_h)\vee 1}$$ (by eq. (26))
$$= \frac{1}{2H}\mathbb{E}_I\left[\bar{V}_{h+B_h}^k(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I)) - V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I))\right] + \frac{4H^2SL_\delta^k}{n_h^k(s,B_h)\vee 1}$$
(by eq. (27))
$$\leq \frac{1}{2H}\left(1+\frac{1}{2H}\right)\mathbb{E}_I\left[\bar{V}_{h+B_h}^k(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I)) - V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I))\right] + \frac{4H^2SL_\delta^k}{n_h^k(s,B_h)\vee 1}.$$

In the last inequality, we slightly increased the coefficient to simplify the final bound – this is again valid because of the optimism, which ensures the positivity of the value difference. Adding this to the bound on the bonus concludes the proof.

Bounding term (ii)**.** Noting that $\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\right] = Q_h^*(s,B_h)$, we can bound term (ii) using event $E^{opt}(k)$,

$$(ii) = \widehat{\mathbb{E}}_{h,s,B_h}^k\left[\mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\right] - \mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)\right] \leq \sqrt{\frac{2\mathrm{Var}_h^*(s,B_h)L_\delta^k}{n_h^k(s,B_h)\vee 1}} + \frac{HL_\delta^k}{n_h^k(s,B_h)\vee 1}.$$

By Lemma 11 w.r.t. $\mathfrak{Q}_h^*(s,B_h;I,V_{h+B_h}^*)$ and $\mathfrak{V}_h^{\pi^k}(s,I)$, we can replace the variance of the optimal Q-value with the variance of the policy π^k. Specifically, choosing $\alpha = 2\sqrt{2L_\delta^k}H$ and $n = n_h^k(s,B_h)\vee 1$, we get

$$(ii) \leq \frac{\sqrt{2\mathrm{Var}_h^{\pi^k}(s)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} + \frac{1}{2H}\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{V}_h^{\pi^k}(s,I)\right] + \frac{2H^2L_\delta^k}{n_h^k(s,B_h)\vee 1}$$
$$\leq \frac{\sqrt{2\mathrm{Var}_h^{\pi^k}(s)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} + \frac{1}{2H}\mathbb{E}_I\left[\mathfrak{Q}_h^*(s,B_h;I,\bar{V}_{h+B_h}^k) - \mathfrak{V}_h^{\pi^k}(s,I)\right] + \frac{2H^2L_\delta^k}{n_h^k(s,B_h)\vee 1}$$ (by eq. (26))
$$= \frac{\sqrt{2\mathrm{Var}_h^{\pi^k}(s)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} + \frac{1}{2H}\mathbb{E}_I\left[\bar{V}_{h+B_h}^k(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I)) - V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I))\right] + \frac{2H^2L_\delta^k}{n_h^k(s,B_h)\vee 1}$$
(by eq. (27))
$$\leq \frac{\sqrt{2\mathrm{Var}_h^{\pi^k}(s)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} + \frac{1}{2H}\left(1+\frac{1}{2H}\right)\mathbb{E}_I\left[\bar{V}_{h+B_h}^k(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I)) - V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I))\right] + \frac{2H^2L_\delta^k}{n_h^k(s,B_h)\vee 1}.$$

Substituting $(i),(ii)$ and $b_h^k(s,B_h)$ back into eq. (25) (and slightly increasing some constants to merge expressions), we get the bound

$$\bar{V}_h^k(s) - V_h^{\pi^k}(s) \le \frac{3\sqrt{2\mathrm{Var}_h^{\pi^k}(s)L_\delta^k}}{\sqrt{n_h^k(s,B_h)\vee 1}} + \left(1+\frac{3}{2H}\right)\left(1+\frac{1}{2H}\right)\mathbb{E}_I\left[\bar{V}_{h+B_h}^k(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I)) - V_{h+B_h}^{\pi^k}(\mathfrak{s}_{h+B_h|h}(s,\phi^{k,h},I))\right] + \frac{46H^2SL_\delta^k}{n_h^k(s,B_h)\vee 1}. \tag{28}$$

The next step is to recursively apply this value difference formula between timesteps and episodes. Specifically, for $i\in[H]$, we apply the recursion with $h\leftarrow \mathfrak{t}_i^k$; we remind that this is the step in which the i^{th} batch started on episode k (with $\mathfrak{t}_1^k=1$). In particular, if $\mathfrak{t}_i^k\le H$, then $\mathfrak{t}_{i+1}^k=\mathfrak{t}_i^k+B_{\mathfrak{t}_i^k}^k$, and we use the convention that $\mathfrak{t}_i^k=H+1$ if there were fewer than i batches. Notably, since we look at batch starts, new lookahead information is independently sampled from $\mathcal{I}$, so the expectation w.r.t. $\mathcal{I}$ coincides with the sampling of the lookahead during the interaction (given the batch starting time, state and policy). Then, it holds that

$$\begin{aligned}
&\sum_{k=1}^K\left(\bar{V}_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k) - V_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)\right)\\
&\quad\le \sum_{k=1}^K \frac{3\sqrt{2\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)L_\delta^k}}{\sqrt{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k,B_{\mathfrak{t}_i^k}^k)\vee 1}} + \left(1+\frac{3}{2H}\right)\left(1+\frac{1}{2H}\right)\sum_{k=1}^K\mathbb{E}\left[\bar{V}_{\mathfrak{t}_{i+1}^k}^k(\mathfrak{s}_{\mathfrak{t}_{i+1}^k|\mathfrak{t}_i^k}(s_{\mathfrak{t}_i^k}^k,\phi^{k,\mathfrak{t}_i^k},I_{\mathfrak{t}_i^k}^k)) - V_{\mathfrak{t}_{i+1}^k}^{\pi^k}(\mathfrak{s}_{\mathfrak{t}_{i+1}^k|\mathfrak{t}_i^k}(s_{\mathfrak{t}_i^k}^k,\phi^{k,\mathfrak{t}_i^k},I_{\mathfrak{t}_i^k}^k))|\mathfrak{t}_i^k,s_{\mathfrak{t}_i^k}^k,\pi^k\right]\\
&\qquad+\sum_{k=1}^K\frac{46H^2SL_\delta^k}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k,B_{\mathfrak{t}_i^k}^k)\vee 1}\\
&\quad\overset{(1)}{=}\sum_{k=1}^K \frac{3\sqrt{2\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)L_\delta^k}}{\sqrt{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k,B_{\mathfrak{t}_i^k}^k)\vee 1}} + \left(1+\frac{3}{2H}\right)\left(1+\frac{1}{2H}\right)\sum_{k=1}^K\mathbb{E}\left[\bar{V}_{\mathfrak{t}_{i+1}^k}^k(s_{\mathfrak{t}_{i+1}^k}^k) - V_{\mathfrak{t}_{i+1}^k}^{\pi^k}(s_{\mathfrak{t}_{i+1}^k}^k)|\mathfrak{t}_i^k,s_{\mathfrak{t}_i^k}^k,\pi^k\right] + \sum_{k=1}^K\frac{46H^2SL_\delta^k}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k,B_{\mathfrak{t}_i^k}^k)\vee 1}\\
&\quad\overset{(2)}{\le}\sum_{k=1}^K \frac{3\sqrt{2\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)L_\delta^k}}{\sqrt{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k,B_{\mathfrak{t}_i^k}^k)\vee 1}} + \left(1+\frac{3}{2H}\right)\left(1+\frac{1}{2H}\right)^2\sum_{k=1}^K\left(\bar{V}_{\mathfrak{t}_{i+1}^k}^k(s_{\mathfrak{t}_{i+1}^k}^k) - V_{\mathfrak{t}_{i+1}^k}^{\pi^k}(s_{\mathfrak{t}_{i+1}^k}^k)\right) + 90H^2L_\delta^K + \sum_{k=1}^K\frac{46H^2SL_\delta^k}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k,B_{\mathfrak{t}_i^k}^k)\vee 1},
\end{aligned}\tag{29}$$

where (1) is since $\mathfrak{s}_{\mathfrak{t}_{i+1}^k|\mathfrak{t}_i^k}(s_{\mathfrak{t}_i^k}^k,\phi^{k,\mathfrak{t}_i^k},I_{\mathfrak{t}_i^k}^k)$ points to the exact state the agent will reach in step $\mathfrak{t}_{i+1}^k$, and (2) is under E^{diff} (with a slight increase to the log-term). We now apply this recursion $H-1$ times, from $\mathfrak{t}_1^k=1$ up to $\mathfrak{t}_H^k$ (which is trivially $H+1$ if there is no such batch), and under the convention that $n_{H+1}^k(s,B)=\infty$. At the end of the recursion, both values equal zero, and so we get under $\mathbb{G}$ that

$$\begin{aligned}
&\sum_{k=1}^K\left(\bar{V}_1^k(s_1^k) - V_1^{\pi^k}(s_1^k)\right)\\
&\quad=\sum_{k=1}^K\left(\bar{V}_{\mathfrak{t}_1^k}^k(s_{\mathfrak{t}_1^k}^k) - V_{\mathfrak{t}_1^k}^{\pi^k}(s_{\mathfrak{t}_1^k}^k)\right)\\
&\quad\le\left(1+\frac{3}{2H}\right)^H\left(1+\frac{1}{2H}\right)^{2H}\sum_{k=1}^K\sum_{i=1}^H\frac{3\sqrt{2\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)L_\delta^k}}{\sqrt{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k,B_{\mathfrak{t}_i^k}^k)\vee 1}} + \left(1+\frac{3}{2H}\right)^H\left(1+\frac{1}{2H}\right)^{2H}\left(\sum_{k=1}^K\sum_{i=1}^H\frac{46H^2SL_\delta^k}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k,B_{\mathfrak{t}_i^k}^k)\vee 1} + 90H^3L_\delta^K\right)\\
&\quad\le e^3\sum_{k=1}^K\sum_{i=1}^H\frac{3\sqrt{2\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)L_\delta^k}}{\sqrt{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k,B_{\mathfrak{t}_i^k}^k)\vee 1}} + e^3\left(\sum_{k=1}^K\sum_{i=1}^H\frac{46H^2SL_\delta^k}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k,B_{\mathfrak{t}_i^k}^k)\vee 1} + 90H^3L_\delta^K\right)\\
&\quad\le 3e^3\left(2\sqrt{2SH^3K\ell}L_\delta^K + 4\sqrt{S\ell}H^2(L_\delta^K)^{1.5}\right) + e^3(SH\ell(2+\ln(K)))\big(136H^2S\big)L_\delta^k \qquad\text{(by Lemmas 7 and 16)}\\
&\quad=\mathcal{O}\left(\sqrt{SH^3K\ell}L_\delta^K + H^3S^2\ell\left(L_\delta^K\right)^2\right).
\end{aligned}\tag{30}$$

Recalling that the good event holds w.p. $1-\delta$ (by Lemma 5) and using the optimistic bound $V_h^*(s) - V_h^{\pi^k}(s) \le \bar{V}_h^k(s) - V_h^{\pi^k}(s)$ (by Lemma 4) yields that w.p. $1-\delta$, for all $K \ge 1$,

$$\mathrm{Reg}(K) = \sum_{k=1}^{K}\Big(V_1^*(s_1^k) - V_1^{\pi^k}(s_1^k)\Big) \le \sum_{k=1}^{K}\Big(\bar{V}_1^k(s_1^k) - V_1^{\pi^k}(s_1^k)\Big) = \mathcal{O}\Big(\sqrt{SH^3K\ell}L_\delta^K + H^3S^2\ell\big(L_\delta^K\big)^2\Big).$$

This concludes the proof. □

E.1 Regret When Lookahead Is Obtained for All Batching Horizons

We now briefly discuss the case in which we obtain full ℓ-step lookahead at the beginning of each batch (but still compete against the best ABP). This additional information is extremely beneficial for learning: we can update all empirical estimators for all $B \in [\ell_h]$ (instead of just for the played B_h^k).

Concretely, we run the exact same algorithm, but at any step h in which a new batch started, we obtain lookahead $I_{h,B}^k$ for all $B \in [\ell_h]$. In particular, we may update $n_h^k(s_h^k, B) \leftarrow n_h^k(s_h^k, B) + 1$ for all $B \in [\ell_h]$, and so $n_h^k(s,B) = n_h^k(s)$. The algorithm does not use any of the additional observations inside an episode, so all of its properties are retained, and the same concentration bounds hold with the counts $n_h^k(s,B) = n_h^k(s)$, with at most minor changes to the union bounds. Thus, following the same derivation up to eq. (30), and under the good event, it holds that

$$\begin{aligned}\mathrm{Reg}(K) &\le e^3\sum_{k=1}^{K}\sum_{i=1}^{H}\frac{3\sqrt{2\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)L_\delta^k}}{\sqrt{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k, B_{\mathfrak{t}_i^k}^k)\vee 1}} + e^3\left(\sum_{k=1}^{K}\sum_{i=1}^{H}\frac{46H^2SL_\delta^k}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k, B_{\mathfrak{t}_i^k}^k)\vee 1} + 90H^2L_\delta^k\right)\\ &= e^3\sum_{k=1}^{K}\sum_{i=1}^{H}\frac{3\sqrt{2\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)L_\delta^k}}{\sqrt{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k)\vee 1}} + e^3\left(\sum_{k=1}^{K}\sum_{i=1}^{H}\frac{46H^2SL_\delta^k}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k)\vee 1} + 90H^2L_\delta^k\right). \qquad (31)\end{aligned}$$

Importantly, we can follow the standard derivation of the inverse count sums (e.g., Lemma 16) and get

$$\begin{aligned}\sum_{k=1}^{K}\sum_{i=1}^{H}\frac{1}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k)\vee 1} &= \sum_{k=1}^{K}\sum_{h=1}^{H}\sum_{s\in\mathcal{S}}\frac{\mathbb{1}\big\{\exists i\in[H]:\mathfrak{t}_i^k = h\big\}\mathbb{1}\big\{s_h^k = s\big\}}{n_h^k(s)\vee 1} = \sum_{h=1}^{H}\sum_{s\in\mathcal{S}}\sum_{m=0}^{n_h^K(s)}\frac{1}{m\vee 1}\\ &\le \sum_{h=1}^{H}\sum_{s\in\mathcal{S}}\big(2+\ln\big(n_h^K(s)\vee 1\big)\big) \le \sum_{h=1}^{H}\sum_{s\in\mathcal{S}}(2+\ln(K)) \le SH(2+\ln(K)).\end{aligned} \qquad (32)$$

This relation can also be used to adapt Lemma 7 and get

$$\begin{aligned}\sum_{k=1}^{K}\sum_{i=1}^{H}\frac{\sqrt{\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)}}{\sqrt{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k)\vee 1}} &\le \sqrt{\sum_{k=1}^{K}\sum_{i=1}^{H}\frac{1}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k)\vee 1}}\sqrt{\sum_{k=1}^{K}\sum_{i=1}^{H}\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)} && \text{(Cauchy Schwarz)}\\ &\le \sqrt{SH(2+\ln(K))}\sqrt{\sum_{k=1}^{K}\sum_{i=1}^{H}\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)} && \text{(by eq. (32))}\\ &\le \sqrt{2SHL_\delta^K}\sqrt{2\sum_{k=1}^{K}\sum_{i=1}^{H}\mathbb{E}\Big[\mathrm{Var}_{\mathfrak{t}_i^k}^{\pi^k}(s_{\mathfrak{t}_i^k}^k)|s_1^k,\pi^k\Big] + 4H^3\ln\frac{4K(K+1)}{\delta}} && \text{(under } E^{\mathrm{Var}}\text{)}\\ &\le \sqrt{2SHL_\delta^K}\sqrt{2KH^2 + 4H^3\ln\frac{4K(K+1)}{\delta}} && \text{(by Lemma 6)}\\ &\le 2\sqrt{SH^3KL_\delta^K} + \sqrt{8S}H^2L_\delta^K.\end{aligned}$$

Substituting back into eq. (31) results in a regret bound of $\tilde{\mathcal{O}}\Big(\sqrt{H^3SK}\Big)$. Following the same flow will similarly lead to this regret bound for fixed batching policies (with any predefined batching horizon).

E.2 Lemmas for Bounding Variance Terms

First, we prove a variant of the law of total variance (LTV) for ABPs.

Lemma 6. *Let π be an ABP, and let $\mathfrak{t}_i^\pi$ be the (random) step in which π starts its i^{th} batch. Finally, denote the variance of π w.r.t. the immediate lookahead information by*

$$\mathrm{Var}_h^\pi(s) = \mathbb{E}\left[\left(\sum_{t=h}^{h+B_h-1} R_t + V_{h+B_h}^\pi(s_{h+B_h}) - V_h^\pi(s)\right)^2 |\pi, s_h = s, E_h^n\right].$$

Then, the following holds:

$$\mathbb{E}\left[\sum_{i=1}^{H} \mathrm{Var}_{\mathfrak{t}_i^\pi}^\pi(s_{\mathfrak{t}_i^\pi})|\pi, s_1 = s\right] = \mathbb{E}\left[\left(\sum_{t=1}^{H} R_t - V_1^\pi(s_1)\right)^2 |\pi, s_1 = s\right] \le H^2.$$

Proof. We extend and adapt the proof of [27]. We start by proving that for any $h \in [H]$, it holds that

$$\mathbb{E}\left[\left(\sum_{t=h}^{H} R_t - V_h^\pi(s_h)\right)^2 |\pi, s_h, E_h^n\right] = \mathrm{Var}_h^\pi(s_h) + \mathbb{E}\left[\left(\sum_{t=h+B_h}^{H} R_t - V_{h+B_h}^\pi(s_{h+B_h})\right)^2 |\pi, s_h, E_h^n\right]. \tag{33}$$

Starting from the l.h.s., we have

$$\begin{aligned}
&\mathbb{E}\left[\left(\sum_{t=h}^{H} R_t - V_h^\pi(s_h)\right)^2 |\pi, s_h, E_h^n\right] \\
&= \mathbb{E}\left[\left(\left(\sum_{t=h}^{h+B_h-1} R_t + V_{h+B_h}^\pi(s_{h+B_h}) - V_h^\pi(s_h)\right) + \left(\sum_{t=h+B_h}^{H} R_t - V_{h+B_h}^\pi(s_{h+B_h})\right)\right)^2 |\pi, s_h, E_h^n\right] \\
&= \mathbb{E}\left[\left(\sum_{t=h}^{h+B_h-1} R_t + V_{h+B_h}^\pi(s_{h+B_h}) - V_h^\pi(s_h)\right)^2 |\pi, s_h, E_h^n\right] + \mathbb{E}\left[\left(\sum_{t=h+B_h}^{H} R_t - V_{h+B_h}^\pi(s_{h+B_h})\right)^2 |\pi, s_h, E_h^n\right] \\
&\quad + 2\mathbb{E}\left[\left(\sum_{t=h}^{h+B_h-1} R_t + V_{h+B_h}^\pi(s_{h+B_h}) - V_h^\pi(s_h)\right)\left(\sum_{t=h+B_h}^{H} R_t - V_{h+B_h}^\pi(s_{h+B_h})\right)|\pi, s_h, E_h^n\right] \\
&= \mathrm{Var}_h^\pi(s_h) + \mathbb{E}\left[\left(\sum_{t=h+B_h}^{H} R_t - V_{h+B_h}^\pi(s_{h+B_h})\right)^2 |\pi, s_h, E_h^n\right] \\
&\quad + \underbrace{2\,\mathbb{E}\left[\left(\sum_{t=h}^{h+B_h-1} R_t + V_{h+B_h}^\pi(s_{h+B_h}) - V_h^\pi(s_h)\right)\left(\sum_{t=h+B_h}^{H} R_t - V_{h+B_h}^\pi(s_{h+B_h})\right)|\pi, s_h, E_h^n\right]}_{(*)}
\end{aligned}$$

Therefore, to prove eq. (33), we only need to prove that the cross-term $(*)$ equals zero. By the batching structure, given E_h^n and a policy π, the starting point of the next point is deterministically determined by s_h to be $h + B_h$. Therefore,

$$\begin{aligned}
(*) &= \mathbb{E}\left[\left(\sum_{t=h}^{h+B_h-1} R_t + V_{h+B_h}^\pi(s_{h+B_h}) - V_h^\pi(s_h)\right)\left(\sum_{t=h+B_h}^{H} R_t - V_{h+B_h}^\pi(s_{h+B_h})\right)|\pi, s_h, E_h^n, E_{h+B_h}^n\right] \\
&\overset{(1)}{=} \mathbb{E}\Bigg\{\left(\sum_{t=h}^{h+B_h-1} R_t + V_{h+B_h}^\pi(s_{h+B_h}) - V_h^\pi(s_h)\right) \\
&\qquad \cdot \underbrace{\mathbb{E}\left[\sum_{t=h+B_h}^{H} R_t - V_{h+B_h}^\pi(s_{h+B_h})|\pi, s_h, E_h^n, E_{h+B_h}^n, s_{h+B_h}, R_h, \ldots, R_{h+B_h-1}\right]}_{=0} |\pi, s_h, E_h^n, E_{h+B_h}^n\Bigg\} \\
&\overset{(2)}{=} 0
\end{aligned}$$

Relation (1) is by the tower rule, and (2) is by the definition of the value as the future cumulative reward given that a new batch starts. In particular, notice that given B_h (indicating the time $h + B_h$), $E^n_{h+B_h}$ and s_{h+B_h}, the rewards collected by an ABP are independent of all past events due to the batching of the lookahead. This concludes the proof of eq. (33).

We now utilize this equality to prove the main result. In particular, if h is the timestep in which the i^{th} batch has started at ($\mathfrak{t}^\pi_i = h$), we have that $\mathfrak{t}^\pi_{i+1} = h + B_h$, and the following recursion

$$\begin{aligned}
&\mathbb{E}\left[\left(\sum_{t=\mathfrak{t}^\pi_i}^{H} R_t - V^\pi_{\mathfrak{t}^\pi_i}(s_{\mathfrak{t}^\pi_i})\right)^2 |\pi, \mathfrak{t}^\pi_i, s_{\mathfrak{t}^\pi_i}\right] \\
&= \mathrm{Var}^\pi_{\mathfrak{t}^\pi_i}(s_{\mathfrak{t}^\pi_i}) + \mathbb{E}\left[\left(\sum_{t=\mathfrak{t}^\pi_{i+1}}^{H} R_t - V^\pi_{\mathfrak{t}^\pi_{i+1}}(s_{\mathfrak{t}^\pi_{i+1}})\right)^2 |\pi, \mathfrak{t}^\pi_i, s_{\mathfrak{t}^\pi_i}\right] && \text{(By eq. (33), } \mathfrak{t}^\pi_{i+1} = \mathfrak{t}^\pi_i + B_{\mathfrak{t}^\pi_i}) \\
&= \mathrm{Var}^\pi_{\mathfrak{t}^\pi_i}(s_{\mathfrak{t}^\pi_i}) + \mathbb{E}\left[\mathbb{E}\left[\left(\sum_{t=\mathfrak{t}^\pi_{i+1}}^{H} R_t - V^\pi_{\mathfrak{t}^\pi_{i+1}}(s_{\mathfrak{t}^\pi_{i+1}})\right)^2 |\pi, \mathfrak{t}^\pi_i, s_{\mathfrak{t}^\pi_i}, \mathfrak{t}^\pi_{i+1}, s_{\mathfrak{t}^\pi_{i+1}}\right] |\pi, \mathfrak{t}^\pi_i, s_{\mathfrak{t}^\pi_i}\right] && \text{(tower rule)} \\
&\overset{(1)}{=} \mathrm{Var}^\pi_{\mathfrak{t}^\pi_i}(s_{\mathfrak{t}^\pi_i}) + \mathbb{E}\left[\mathbb{E}\left[\left(\sum_{t=\mathfrak{t}^\pi_{i+1}}^{H} R_t - V^\pi_{\mathfrak{t}^\pi_{i+1}}(s_{\mathfrak{t}^\pi_{i+1}})\right)^2 |\pi, \mathfrak{t}^\pi_{i+1}, s_{\mathfrak{t}^\pi_{i+1}}\right] |\pi, \mathfrak{t}^\pi_i, s_{\mathfrak{t}^\pi_i}\right] \\
&\overset{(2)}{=} \mathrm{Var}^\pi_{\mathfrak{t}^\pi_i}(s_{\mathfrak{t}^\pi_i}) + \mathbb{E}\left[\sum_{j=i+1}^{H} \mathrm{Var}^\pi_{\mathfrak{t}^\pi_j}(s_{\mathfrak{t}^\pi_j})|\pi, \mathfrak{t}^\pi_i, s_{\mathfrak{t}^\pi_i}\right],
\end{aligned}$$

where (1) is since all events at $t \geq \mathfrak{t}^\pi_{i+1}$ are independent of events at previous times given $\mathfrak{t}^\pi_{i+1}, s_{\mathfrak{t}^\pi_{i+1}}$ (due to the start of a new batch), and (2) recursively applies the same relation up to the potentially H^{th} batch (noting that when $\mathfrak{t}^\pi_j = H + 1$, all terms are zero). Finally, recalling that $\mathfrak{t}^\pi_1 = 1$, we get the desired identity

$$\mathbb{E}\left[\sum_{i=1}^{H} \mathrm{Var}^\pi_{\mathfrak{t}^\pi_i}(s_{\mathfrak{t}^\pi_i})|\pi, s_1 = s\right] = \mathbb{E}\left[\left(\sum_{t=1}^{H} R_t - V^\pi_1(s_1)\right)^2 |\pi, s_1 = s\right].$$

The r.h.s. is trivially bounded by H^2, because $R_h \in [0, 1]$ for all h and $V^\pi_1(s_1) \in [0, H]$. □

Lemma 7. *Under the event E^{Var} it holds that*

$$\sum_{k=1}^{K}\sum_{i=1}^{H} \frac{\sqrt{\mathrm{Var}^{\pi^k}_{\mathfrak{t}^k_i}(s^k_{\mathfrak{t}^k_i})}}{\sqrt{n^k_{\mathfrak{t}^k_i}(s^k_{\mathfrak{t}^k_i}, B^k_{\mathfrak{t}^k_i}) \vee 1}} \leq 2\sqrt{SH^3K\ell L^K_\delta} + \sqrt{8S\ell}H^2L^K_\delta.$$

Proof.

$$\begin{aligned}
\sum_{k=1}^{K}\sum_{i=1}^{H} \frac{\sqrt{\mathrm{Var}^{\pi^k}_{\mathfrak{t}^k_i}(s^k_{\mathfrak{t}^k_i})}}{\sqrt{n^k_{\mathfrak{t}^k_i}(s^k_{\mathfrak{t}^k_i}, B^k_{\mathfrak{t}^k_i}) \vee 1}} &\leq \sqrt{\sum_{k=1}^{K}\sum_{i=1}^{H} \frac{1}{n^k_{\mathfrak{t}^k_i}(s^k_{\mathfrak{t}^k_i}, B^k_{\mathfrak{t}^k_i}) \vee 1}} \sqrt{\sum_{k=1}^{K}\sum_{i=1}^{H} \mathrm{Var}^{\pi^k}_{\mathfrak{t}^k_i}(s^k_{\mathfrak{t}^k_i})} \\
&&& \text{(Cauchy Schwarz)} \\
&\leq \sqrt{SH\ell(2 + \ln(K))} \sqrt{\sum_{k=1}^{K}\sum_{i=1}^{H} \mathrm{Var}^{\pi^k}_{\mathfrak{t}^k_i}(s^k_{\mathfrak{t}^k_i})} && \text{(by lemma 16)} \\
&\leq \sqrt{2SH\ell L^K_\delta} \sqrt{2\sum_{k=1}^{K}\sum_{i=1}^{H} \mathbb{E}\left[\mathrm{Var}^{\pi^k}_{\mathfrak{t}^k_i}(s^k_{\mathfrak{t}^k_i})|s^k_1, \pi^k\right] + 4H^3 \ln\frac{4K(K+1)}{\delta}} \\
&&& \text{(under } E^{\mathrm{Var}}) \\
&\leq \sqrt{2SH\ell L^K_\delta} \sqrt{2KH^2 + 4H^3 \ln\frac{4K(K+1)}{\delta}} && \text{(by Lemma 6)} \\
&\leq 2\sqrt{SH^3K\ell L^K_\delta} + \sqrt{8S\ell}H^2L^K_\delta.
\end{aligned}$$

□

F Computational Complexity

In this short appendix, we analyze the computational complexity of our AL-UCB.

We separate the computational analysis into two components: value calculations and the execution of the policy π^k. For both, we rely on $\mathfrak{J}^I_{h,b}(s,s')$ and $\mathfrak{S}^I_{h,b}(s)$. Given s and I, both can be calculated for all s' and $b \leq B$ in $\mathcal{O}(SAB)$ time using the dynamic programming equations in eq. (9). We can also store the action that optimally leads to each s' in $\mathfrak{J}^I_{h,b}(s,s')$, using SB memory: this permits reconstructing the optimal path from the initial state to any s'. Thus, the overall memory complexity given the initial s and I is $\mathcal{O}(SB)$

Policy Execution. Assume we pre-computed the optimistic values $\bar{V}^k_h$. For any step h in which a batch starts, we compute $\mathfrak{J}^k_h, \mathfrak{S}^k_h$, the values of $\mathfrak{J}^I_{h,b}(s,s')$ and $\mathfrak{S}^I_{h,b}(s)$ starting from s^k_h up to range $B = B^k_h$. Recalling that

$$\mathfrak{Q}^*_h(s,B;I,V) = \max_{s' \in \mathfrak{S}^I_{h,B}(s)} \left\{ \mathfrak{J}^I_{h,B}(s,s') + V(s') \right\} = \max_{\phi \in \Pi_{det}} \left\{ \sum_{t=h}^{h+B-1} \mathfrak{R}_{t|h}(s,\phi,I) + V(\mathfrak{s}_{h+B|h}(s,\phi,I)) \right\},$$

we observe that the agent plans $\phi^{k,h}$ to reach $s' \in \mathfrak{S}^k_h$ for which $\mathfrak{J}^k_h(s^k_h,s') + \bar{V}^k_{h+B}(s')$ is maximized.

Computing $\mathfrak{J}^k_h, \mathfrak{S}^k_h$ and $\phi^{k,h}$ (using the action trace) requires $\mathcal{O}(SAB^k_h)$ computations and $\mathcal{O}(SB^k_h)$ memory. At the end of the episode, we store $\mathfrak{J}^k_h, \mathfrak{S}^k_h$, so the per-episode computational complexity is $\mathcal{O}(SAH)$, and we store a memory of at most $\mathcal{O}(SH)$.

Value Computations. For computing the optimistic values, we need to compute $\bar{Q}^k_h(s,B)$ for all h, s, B. To this end, we use the precomputed $\mathfrak{J}^k_h, \mathfrak{S}^k_h$ from previous episodes. Specifically, we need to compute empirical expectation w.r.t. past episodes. For each (h,s), we have at most k past observations (divided across all batching horizons), so each empirical expectation contains at most k terms. Each term requires $\mathcal{O}(S)$ computations, so the per-episode planning complexity is $\mathcal{O}(S^2Hk)$. Overall, the per-episode computational complexity is $\mathcal{O}(SAH + S^2Hk)$, and the memory up to episode k is $\mathcal{O}(SHk)$.

G Concentration Results

Lemma 8. *Fix $K, n \in \mathbb{N}, C > 0$ and let $X \in [0,C]^K, Y \subseteq [K]$ be two not necessarily independent random variables (with $Y \neq \emptyset$ w.p. 1). Also, for any $V, V_0 \in \mathbb{R}^K$, define the functions*

$$f(X,Y;V) = \max_{k \in Y}\{X(k) + V(k)\}, \qquad \Delta f(X,Y;V,V_0) = f(X,Y;V) - f(X,Y;V_0).$$

Finally, let $\{X_i, Y_i\}_{i\in[n]}$ be n i.i.d. samples from the joint distribution of (X,Y). Then, for any $\delta \in (0,1)$, $V_0 \in \mathbb{R}^K$ and $\alpha \in (0,1]$, it holds w.p. at least $1-\delta$ that

$$\forall V \in \mathbb{R}^K \; s.t. \; V - V_0 \in [0,C]^K: \quad \frac{1}{n}\sum_{i=1}^{n} \Delta f(X_i,Y_i;V,V_0) \leq (1+\alpha)\mathbb{E}[\Delta f(X,Y;V,V_0)] + \frac{2CK\ln\frac{3n}{\delta}}{\alpha n}. \tag{34}$$

Proof. As a preliminary step, given a fixed $V \in \mathbb{R}^K$ s.t. $V - V_0 \in [0,C]^K$, we define the random variables $Z_i^V = \Delta f(X_i,Y_i;V,V_0)$ and analyze their concentration properties. For ease of notation, we denote $m^V = \mathbb{E}\big[Z_i^V\big]$ and

$$V_0 + [0,C]^K \triangleq \big\{V \in \mathbb{R}^K : \forall k \in [K], V(k) \in [V_0(k), V_0(k)+C]\big\}.$$

Then, eq. (34) can be rewritten as

$$\forall V \in V_0 + [0,C]^K, \qquad \frac{1}{n}\sum_{i=1}^{n} Z_i^V \leq (1+\alpha)m^V + \frac{2CK\ln\frac{3n}{\delta}}{\alpha n}.$$

The support of Z_i^V. Since $V(k) \geq V_0(k)$ for all $k \in [K]$, then for any $X \in \mathbb{R}^K, Y \subseteq [K]$

$$f(X,Y;V) = \max_{k\in Y}\{X(k)+V(k)\} \geq \max_{k\in Y}\{X(k)+V_0(k)\} = f(X,Y;V_0)$$

and so $Z_i^V \geq 0$. Moreover, it holds that

$$\begin{aligned} Z_i^V &= \max_{k\in Y_i}\{X_i(k)+V(k)\} - \max_{k\in Y_i}\{X_i(k)+V_0(k)\} \\ &\leq \max_{k\in Y_i}\{(X_i(k)+V(k)) - (X_i(k)+V_0(k))\} \\ &\leq \max_{k\in[K]}\{V(k)-V_0(k)\} \\ &\leq C. \end{aligned}$$

In other words, it holds that $Z_i^V \in [0,C]$, and so does $m^V \in [0,C]$.

Next, by Lemma 9 applied on $Z_i^V - m^V \in [-C,C]$, for any $\eta \in \big(0, \frac{1}{C}\big]$ and $\delta' > 0$, it holds w.p. at least $1-\delta'$ that

$$\begin{aligned} \sum_{i=1}^{n} Z_i^V &\leq n\cdot m^V + \eta\sum_{i=1}^{n}\mathbb{E}\Big[\big(Z_i^V - m^V\big)^2\Big] + \frac{\ln\frac{1}{\delta'}}{\eta} \\ &\leq n\cdot m^V + \eta\sum_{i=1}^{n}\mathbb{E}\Big[\big(Z_i^V\big)^2\Big] + \frac{\ln\frac{1}{\delta'}}{\eta} \\ &\leq n\cdot m^V + C\eta\sum_{i=1}^{n}\mathbb{E}\big[Z_i^V\big] + \frac{\ln\frac{1}{\delta'}}{\eta} && (Z_i^V \in [0,C]) \\ &= n(1+C\eta)m^V + \frac{\ln\frac{1}{\delta'}}{\eta}. \end{aligned}$$

Alternatively, denoting $\alpha = C\eta$, for any $\alpha \in (0,1]$, it holds w.p. $1-\delta'$ that

$$\frac{1}{n}\sum_{i=1}^{n} Z_i^V \leq (1+\alpha)m^V + \frac{C\ln\frac{1}{\delta'}}{\alpha n}. \tag{35}$$

Next, for some $\epsilon > 0$ let $V_1, V_2 \in V_0 + [0, C]^K$ s.t. $V_1(k) \le V_2(k) \le V_1(k) + \epsilon$ for all $k \in [K]$. In particular, for all $i \in [n]$, it holds that

$$\begin{aligned}
Z_i^{V_1} &= \max_{k \in Y_i}\{X_i(k) + V_1(k)\} - \max_{k \in Y_i}\{X_i(k) + V_0(k)\} \\
&\le \max_{k \in Y_i}\{X_i(k) + V_2(k)\} - \max_{k \in Y_i}\{X_i(k) + V_0(k)\} && (V_1(k) \le V_2(k) \text{ for all } k) \\
&= Z_i^{V_2} \\
&\le \max_{k \in Y_i}\{X_i(k) + V_1(k) + \epsilon\} - \max_{k \in Y_i}\{X_i(k) + V_0(k)\} && (V_2(k) \le V_1(k) + \epsilon \text{ for all } k) \\
&= \max_{k \in Y_i}\{X_i(k) + V_1(k)\} + \epsilon - \max_{k \in Y_i}\{X_i(k) + V_0(k)\} \\
&= Z_i^{V_1} + \epsilon,
\end{aligned}$$

i.e., $Z_i^{V_1} \le Z_i^{V_2} \le Z_i^{V_1} + \epsilon$. Taking the expectation, this also implies that $m^{V_1} \le m^{V_2} \le m^{V_1} + \epsilon$, and thus

$$\frac{1}{n}\sum_{i=1}^{n} Z_i^{V_2} - (1+\alpha)m^{V_2} \le \frac{1}{n}\sum_{i=1}^{n} Z_i^{V_1} + \epsilon - (1+\alpha)m^{V_1}, \tag{36}$$

We now utilize this Lipschitzness to apply a covering argument. For any $V \in V_0 + [0, C]^K$, let V_ϵ be the rounded version so that each component is rounded down to an element in $\{V_0(k), V_0(k) + \epsilon, \ldots, V_0(k) + (\lceil C/\epsilon \rceil - 1)\epsilon\}$. In particular, each component of V_ϵ lower bounds and is at most ϵ away from the respective component of V. Then, by eq. (36) (with $V \to V_2$ and $V_1 \to V_\epsilon$), for any $\delta' \in (0, 1]$,

$$\begin{aligned}
&\Pr\left\{\exists V \in V_0 + [0, C]^K : \frac{1}{n}\sum_{i=1}^{n} Z_i^V - (1+\alpha)m^V \ge \frac{C \ln \frac{1}{\delta'}}{\alpha n} + \epsilon\right\} \\
&\le \Pr\left\{\exists V \in V_0 + [0, C]^K : \frac{1}{n}\sum_{i=1}^{n} Z_i^{V_\epsilon} + \epsilon - (1+\alpha)m^{V_\epsilon} \ge \frac{C \ln \frac{1}{\delta'}}{\alpha n} + \epsilon\right\} \\
&= \Pr\left\{\exists V_\epsilon \in \mathbb{R}^K \; s.t. \; V_\epsilon - V_0 \in \{0, \epsilon, 2\epsilon, \ldots, (\lceil C/\epsilon \rceil - 1)\epsilon\}^K : \frac{1}{n}\sum_{i=1}^{n} Z_i^{V_\epsilon} - (1+\alpha)m^{V_\epsilon} \ge \frac{C \ln \frac{1}{\delta'}}{\alpha n}\right\} \\
&\le \left(\frac{C}{\epsilon} + 1\right)^K \delta',
\end{aligned}$$

where the last inequality takes the union bound over all possible V_ϵ and applies eq. (35). Finally, for any $\delta \in (0, 1]$ and $\alpha \in (0, 1]$, we fix $\epsilon = \frac{CK \ln \frac{3}{\delta}}{\alpha n}$ and $\delta' = \frac{\delta}{\left(\frac{C}{\epsilon}+1\right)^K}$. Then, w.p., at least $1 - \delta$, for all $V \in V_0 + [0, C]^K$, it holds that

$$\begin{aligned}
\frac{1}{n}\sum_{i=1}^{n} Z_i^V - (1+\alpha)m^V &\le \frac{C \ln \frac{\left(\frac{C}{\epsilon}+1\right)^K}{\delta}}{\alpha n} + \epsilon \\
&= \frac{C \ln \frac{\left(\frac{\alpha n}{K \ln \frac{3}{\delta}}+1\right)^K}{\delta}}{\alpha n} + \frac{CK \ln \frac{3}{\delta}}{\alpha n} \\
&\le \frac{C \ln \frac{(2n)^K}{\delta}}{\alpha n} + \frac{CK \ln \frac{3}{\delta}}{\alpha n} \\
&\le \frac{CK \ln \frac{2n}{\delta}}{\alpha n} + \frac{CK \ln \frac{3}{\delta}}{\alpha n} \\
&\le \frac{2CK \ln \frac{3n}{\delta}}{\alpha n}.
\end{aligned}$$

□

H Existing Auxiliary Lemmas

Lemma 9 (Freedman's inequality, [2], Theorem 1)**.** *Let $\{X_t\}_{t\geq 1}$ be a real-valued martingale difference sequence adapted to a filtration $\{F_t\}_{t\geq 0}$. If $|X_t| \leq R$ a.s. then for any $\eta \in (0, 1/R], T \in \mathbb{N}$ it holds with probability greater than $1-\delta$,*

$$\sum_{t=1}^{T} X_t \leq \eta \sum_{t=1}^{T} \mathbb{E}[X_t^2 | F_{t-1}] + \frac{\log(1/\delta)}{\eta}.$$

We now present a variant of three lemmas from [16]. We remark that in the original statement in [16], all variables were assumed to have finite support. We verified that the proofs, in fact, never rely on this assumption and remain completely identical for continuous random variables, which correspond to the following statements:

Lemma 10 ([16], Lemma 16, event E^{pv2})**.** *Let $X, X_1, \ldots, X_n$ be i.i.d. random variables supported in $[0, C]$ for some $C > 0$. Also define*

$$\widehat{\mathrm{Var}}_n(X) = \frac{1}{n}\sum_{i=1}^{n} X_i^2 - \left(\frac{1}{n}\sum_{i=1}^{n} X_i\right)^2, \quad \textit{and,} \quad \mathrm{Var}(X) = \mathbb{E}[X^2] - (\mathbb{E}[X])^2.$$

Then, for any $\delta \in (0,1)$, w.p. at least $1-\delta$, it holds that :

$$\left|\sqrt{\widehat{\mathrm{Var}}_n(X)} - \sqrt{\mathrm{Var}(X)}\right| \leq 4C\sqrt{\frac{\ln\frac{2}{\delta}}{n \vee 1}}.$$

Lemma 11 ([16], Lemma 21)**.** *Let P be a distribution over $\mathcal{X}$. Also, let $V_1, V_2 : \mathcal{X} \mapsto [0, C]$ be measurable functions for some $C > 0$ such that $V_1(x) \leq V_2(x)$ for all $x \in \mathcal{X}$. Then, for any $\alpha, n > 0$, it holds that*

$$\frac{\sqrt{\mathrm{Var}_P(V_2(X))}}{\sqrt{n}} \leq \frac{\sqrt{\mathrm{Var}_P(V_1(X))}}{\sqrt{n}} + \frac{1}{\alpha}\mathbb{E}_{X\sim P}[V_2(X) - V_1(X)] + \frac{C\alpha}{4n},$$

where $\mathrm{Var}_P(V(X)) = \mathbb{E}_{X\sim P}\left[V(X)^2\right] - \left(\mathbb{E}_{X\sim P}[V(X)]\right)^2$

Lemma 12 ([16], Lemma 22)**.** *Let P, P' be two distributions over $\mathcal{X}$ (w.r.t. the same σ-algebra). Also, let $V_1, V_2, V_3 : \mathcal{X} \mapsto [0, C]$ be measurable functions for some $C > 0$ such that $V_1(x) \leq V_2(x) \leq V_3(x)$ for all $x \in \mathcal{X}$. Finally, assume that*

$$\left|\sqrt{\mathrm{Var}_P(V_2(X))} - \sqrt{\mathrm{Var}_{P'}(V_2(X))}\right| \leq \beta$$

for some $\beta > 0$, where $\mathrm{Var}_P(V(X)) = \mathbb{E}_{X\sim P}\left[V(X)^2\right] - \left(\mathbb{E}_{X\sim P}[V(X)]\right)^2$*. Then, for any $\alpha, n > 0$, it holds that*

$$\frac{\sqrt{\mathrm{Var}_{P'}(V_3(X))}}{\sqrt{n}} \leq \frac{\sqrt{\mathrm{Var}_P(V_1(X))}}{\sqrt{n}} + \frac{1}{\alpha}\mathbb{E}_{X\sim P'}[V_3(X) - V_2(X)] + \frac{1}{\alpha}\mathbb{E}_{X\sim P}[V_2(X) - V_1(X)] + \frac{C\alpha}{2n} + \frac{\beta}{\sqrt{n}}.$$

Lemma 13 ([30], Appendix C.1)**.** *For any $d \in \mathbb{N}$, $p \in \Delta^d$, $v \in \mathbb{R}_+^d$ s.t. $\|v\|_\infty \leq H$, $\delta' \in (0,1)$ and positive integer n, define the function*

$$f(p, v, n) = p^T v + \max\left\{\frac{20}{3}\sqrt{\frac{\mathrm{Var}_p(v)\ln\frac{1}{\delta'}}{n}}, \frac{400}{9}\frac{H\ln\frac{1}{\delta'}}{n}\right\},$$

where $\mathrm{Var}_p(v) = \sum_{i=1}^d p_i v_i^2 - \left(\sum_{i=1}^d p_i v_i\right)^2$*. Then, the function $f(p, v, n)$ is non-decreasing in each entry of v.*

Corollary 14. *For any $d \in \mathbb{N}$, $p \in \Delta^d$, $v \in \mathbb{R}_+^d$ s.t. $\|v\|_\infty \leq H$ and $\alpha \geq 0$, define the function*

$$f(p, v, \alpha) = p^T v + \max\left\{\alpha\sqrt{\mathrm{Var}_p(v)}, \alpha^2 H\right\},$$

where $\mathrm{Var}_p(v) = \sum_{i=1}^d p_i v_i^2 - \left(\sum_{i=1}^d p_i v_i\right)^2$*. Then, the function $f(p, v, \alpha)$ is non-decreasing in each entry of v.*

Proof. The claim trivially holds for $\alpha = 0$. For any $\alpha > 0$, define $\delta' = \exp\{-\frac{9}{400}\alpha^2 n\}$; since $\alpha, n > 0$, this is a valid choice for Lemma 13, as it yields $\delta' \in (0, 1)$. Then, directly substituting this value leads to the $f(p, v, n)$ defined in Lemma 13: since it is non-decreasing in v, so is $f(p, v, \alpha)$ defined in this corollary. □

Lemma 15 ([9], Lemma 27)**.** *Let $\{Y_t\}_{t\geq 1}$ be a real-valued sequence of random variables adapted to a filtration $\{F_t\}_{t\geq 0}$. Assume that for all $t \geq 1$ it holds that $0 \leq Y_t \leq C$ a.s., and let $T \in \mathbb{N}$. Then each of the following inequalities holds with probability greater than $1 - \delta$.*

$$\sum_{t=1}^{T} \mathbb{E}[Y_t|F_{t-1}] \leq \left(1 + \frac{1}{2C}\right) \sum_{t=1}^{T} Y_t + 2(2C+1)^2 \ln \frac{1}{\delta},$$
$$\sum_{t=1}^{T} Y_t \leq 2 \sum_{t=1}^{T} \mathbb{E}[Y_t|F_{t-1}] + 4C \ln \frac{1}{\delta}.$$

Lemma 16 (e.g., [16], Lemma 20)**.** *Under the convention that $n_h^k(s, B) = \infty$ if $h = H + 1$, the following bounds hold:*

$$\sum_{k=1}^{K} \sum_{i=1}^{H} \frac{1}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k, B_{\mathfrak{t}_i^k}^k) \vee 1} \leq SH\ell(2 + \ln(K)).$$

Proof. The claim is standard and is provided due to its simplicity and for completeness. By definition, we can omit from the summation all terms where $\mathfrak{t}_i^k = H + 1$. Since for all $h \in [H]$, there cannot be two batches starting at this step during the same episode, we can write

$$\begin{aligned}
\sum_{k=1}^{K} \sum_{i=1}^{H} \frac{1}{n_{\mathfrak{t}_i^k}^k(s_{\mathfrak{t}_i^k}^k, B_{\mathfrak{t}_i^k}^k) \vee 1} &= \sum_{k=1}^{K} \sum_{h=1}^{H} \sum_{s\in\mathcal{S}} \sum_{B=1}^{\ell_h} \frac{\mathbb{1}\{\exists i \in [H] : \mathfrak{t}_i^k = h\}\mathbb{1}\{s_h^k = s, B_h^k = B\}}{n_h^k(s, B) \vee 1} \\
&\overset{(1)}{=} \sum_{h=1}^{H} \sum_{s\in\mathcal{S}} \sum_{B=1}^{\ell_h} \sum_{m=0}^{n_h^K(s,B)} \frac{1}{m \vee 1} \\
&\overset{(2)}{\leq} \sum_{h=1}^{H} \sum_{s\in\mathcal{S}} \sum_{B=1}^{\ell_h} \big(2 + \ln\big(n_h^K(s, B) \vee 1\big)\big) \\
&\overset{(3)}{\leq} \sum_{h=1}^{H} \sum_{s\in\mathcal{S}} \sum_{B=1}^{\ell_h} (2 + \ln(K)) \\
&\leq SH\ell(2 + \ln(K)).
\end{aligned}$$

Relation (1) holds since the counts start at zero and increase by 1 every time that a batch of size B starts at step h and state s. Relation (2) bounds the harmonic sum. Finally, (3) is since any tuple (h, s, B) can be visited at most once per episode □

I Failure of Existing Approaches

I.1 Fixed Batching Policies

Claim 2.1. *For any $\ell \geq 2, A \geq 2, H \geq \ell + 1$, there exists an environment with $S \leq 1 + A^\ell$ and deterministic transitions s.t. for some fixed initial state, any fixed batching policy collects a value of at most $A^{-\ell/2+1}$ in expectation. In contrast, the optimal ℓ-step lookahead value is larger than $1/2$.*

Proof. Let s_r be the root of a complete tree of depth ℓ with A descendants at each node, where $\ell = 1$ corresponds to a single root node. We fix the transitions to deterministically navigate agents through the tree and then lead to a terminal non-rewarding state. The only rewards are located at the $A^{\ell-1}$ leaves (for each of the A actions) and are independent Bernoulli random variables of mean $A^{-\ell}$. Notably, a lookahead policy at s_r can see whether a unit reward was realized at some leaf and execute a policy that collects it. This strategy obtains a reward of 1 iff such a reward was realized in at least one of the leaves: w.p. at least $1 - (1 - A^{-\ell})^{A^\ell} \geq 1 - 1/e \geq 1/2$. Finally, add an initial state s_1 to be a non-rewarding state that deterministically leads to s_r; starting at this state clearly does not affect the optimal lookahead value. We now show that regardless of the batching horizon B, any fixed batching policy π^B fails.

- If $B \leq \ell/2$, then π^B observes the rewards at a distance of at most B steps, thus limiting to only A^B reachable rewards to which the policy can actively navigate. By the union bound, the probability of a reward being realized in at least one of these leaves is at most $A^B A^{-\ell} \leq A^{-\ell/2}$, so this is the maximal reward that may be collected by such a policy.
- If $B > \ell/2$, then π^B does not observe any information at the first batch and starts the second batch at some node of depth B: from which it can reach at most $A^{\ell-B+1}$ rewards. Thus, the value from this state is upper bounded by the probability of having a reward in at least one of these leaves: at most $A^{\ell-B+1}A^{-\ell} \leq A^{-\ell/2+1}$.

□

I.2 Model Predictive Control

An ℓ-step model predictive control is defined by the set of values $V_h(s)$ for all $h \in [H]$ and $s \in \mathcal{S}$. For any step h, being in a state s and upon observing the ℓ-step lookahead $I_{h,\ell_h}(s)$, the scheme solves for the deterministic policy

$$\phi \in \arg\max_{\phi \in \Pi_{det}} \left\{ \sum_{t=h}^{h+\ell_h-1} \mathfrak{R}_{t|h}(s, \phi, I) + V(\mathfrak{s}_{h+\ell_h|h}(s, \phi, I)) \right\}, \tag{37}$$

and executes the first action in the plan $\phi_h(s)$. Then, after the environment transitions to $s_{h+1} = \mathfrak{s}_{h+1|h}(s, \phi, I)$ and the agent observes an additional step of lookahead, the agent recalculates the plan ϕ and again executes only $\phi_{h+1}(s_{h+1})$. This process repeats until the end of the interaction: planning using the lookahead for ℓ-steps and executing the first action.

We now claim that any backwards-induced value calculation scheme must be exponentially suboptimal. A reasonable definition of a backwards-induced scheme is to assume that the value $V_h(s)$ can only depend on events starting from step h and state s. In fact, for the proof, we will only require a much milder assumption: that if a state s in step h deterministically leads to a non-rewarding terminal state, its value only depends on h and on the reward distribution of the actions. Formally, we say that s_T is a terminal state if for all h, a, $\Pr\{s_{h+1} = s | s_h = s_T, a\} = 1$ and $R_h(s_T, a) = 0$. Then, we say that a state s is a semi-terminal state in step h if $\Pr\{s_{h+1} = s_T | s_h = s, a\} = 1$: it deterministically leads to a terminal state (with potentially random rewards). In particular, we say w.l.o.g. that episodes always reach a terminal state at step $h = H + 1$; therefore, all states are semi-terminal in step $h = H$. We thus assume that the value calculation scheme obeys the following assumption, which is the formal version of Assumption 1:

Assumption 2. Let $\mathcal{D}$ be the set of all (Borel-measurable) distributions over $[0, 1]^A$. For any $h \in [H]$, there exists a function $f_h : \mathcal{D} \mapsto \mathbb{R}_+$ such that if s is a semi-terminal state in step h with a joint reward distribution $\mathcal{R}_h(s) \in \mathcal{D}$ for its actions, it holds that $V_h(s) = f_h(\mathcal{R}_h(s))$.

Claim I.1. *Fix $\ell \geq 2$, $A \geq 2, H \geq \ell + 1$ and let π be an MPC agent whose values are computed by a procedure that follows Assumption 2. Then, there exists an environment with $S \leq A^\ell + \ell + 1$ and deterministic transitions for which $\frac{V_1^\pi(s_1)}{V_1^*(s_1)} \leq A^{1-\ell/2}$, where $V^*(s_1)$ is the value of the optimal ℓ-step lookahead policies. In particular, for the constructed environments, there exists an MPC agent that achieves the optimal value $V_1^*(s_1)$.*

Proof. We create two 'state types' s^B, s^D that will be semi-terminal states in step $\ell + 1$. In states of type s^D, all actions yield a deterministic reward of $A^{-\ell/2}$ and transition to a terminal state s_T, while in states of type s^B, the rewards are independent Bernoulli $\mathrm{Ber}\left(A^{-\ell}\right)$. By Assumption 2, there exist two values V^B, V^D, such that for all states s of type s^B, it holds that $V_{\ell+1}(s) = V^B$, and for all states s of type s^D, it holds that $V_{\ell+1}(s) = V^D$.

We now describe the environments in which we will analyze their value. The initial state is s_1, and it deterministically leads (via two arbitrary actions) to states s^{Tr}, s^L with respective deterministic rewards R^{Tr}, R^L. From s^{Tr}, the environment is a complete A-ary tree of depth ℓ; in particular, it requires less than $A^\ell - 1$ states and has $A^{\ell-1}$ leaves (with A actions each). There are no rewards in the middle of the tree, and the leaves will be either of type s^B or s^D. On the other hand, from s^L, the environment forms a line of length ℓ where the only rewarding state is the last one (of type s^B or s^D). After traversing either the tree or line, the environment leads to a terminal state s^T: this environment can be represented using at most $A^\ell + \ell + 1$ states. We now fix the reward configuration depending on the following cases:

- If $V^B > V^D$: Set $R^{Tr} = R^L = 0$, and put the state s^B in the end of the line and s^D in the leaves of the tree. Since there are no rewards until step $\ell + 1$, the MPC agent will go to the line, which has a higher value $V_{\ell+1}(s^B) = V^B$, and will collect a reward of at most $V_1^\pi(s_1) \leq A \cdot A^{-\ell} = A^{-\ell+1}$. However, by going to the tree, any policy will earn a reward of $A^{-\ell/2}$; therefore, the optimal value is $V_1^*(s_1) = A^{-\ell/2}$ and $\frac{V_1^\pi(s_1)}{V_1^*(s_1)} \leq A^{1-\ell/2}$.
- If $V^D > V^B$: Set $R^{Tr} = R^L = 0$, and put the state s^D in the end of the line and s^B in the leaves of the tree. The agent π will thus navigate to the line and collect a reward of $A^{-\ell/2}$. However, upon reaching the root of the tree, all the rewards in all of the leaves and their actions are revealed. In particular, it is optimal to navigate to any leaf that has a realized unit reward, and the probability it happens is $1 - (1 - A^{-\ell})^{A^\ell} \geq 1 - 1/e$. In particular, this value is higher than $A^{-\ell/2}$ for $A, \ell \geq 2$, which makes this choice optimal. Therefore, it holds that $\frac{V_1^\pi(s_1)}{V_1^*(s_1)} \leq 2A^{-\ell/2} \leq A^{1-\ell/2}$.
- If $V^D = V^B$: we now set $R^{Tr} = 0$ and $R^L = 0.25A^{-\ell/2}$, and again put the state s^D in the end of the line and s^B in the leaves of the tree. Since the values are equal, the reward at the beginning will make the MPC agent choose the line, leading to a reward of $1.25A^{-\ell/2}$. Yet, traversing the tree will still lead to a reward of at least $1 - 1/e$, and so it is still optimal (for $A, \ell \geq 2$) and $\frac{V_1^\pi(s_1)}{V_1^*(s_1)} \leq 2A^{-\ell/2} \leq A^{1-\ell/2}$.

Combined, we get that regardless of the values V^B, V^D, there exists an environment for which $\frac{V_1^\pi(s_1)}{V_1^*(s_1)} \leq A^{1-\ell/2}$. Moreover, in all three cases, the optimal policy decides to traverse the tree and not the line: this behavior can be achieved using an MPC agent by putting a unit value at the leaves of the tree and a zero value at the end of the line: there exists an optimal MPC agent that collects the maximal value. □

I.3 Numerical Illustrations

We end this appendix with two tables that include the numerical values for the examples in Claim 2.1 and Claim 2.2, based on exact closed-form formulas for the analyzed environments.

- Fixed batching counterexample (Claim 2.1): For the tree construction, $S = 2+(A^\ell-1)/(A-1)$ (including the absorbing state), and one best fixed batch length is $B^\star = \lfloor \ell/2 \rfloor + 1$. For this choice of $B^\star$, the exact fixed-batching and optimal values are given by the formulas

$$V_{\mathrm{FB}} = 1 - (1 - A^{-\ell})^{A^{\lceil \ell/2 \rceil}} \qquad \text{and} \qquad V_{\mathrm{opt}} = 1 - (1 - A^{-\ell})^{A^\ell}.$$

  The displayed values are rounded. The values for $A \in \{2, 5\}$ and $\ell \in \{2, 5, 10\}$ are presented in Table 1.

- MPC counterexample (Claim I.1): Taking into account both the tree and line (and a terminal state), it holds that $S = 2 + (A^\ell - 1)/(A - 1) + \ell$. For the illustration, we use the expected reward as the MPC terminal-value rule $V_h(s) = \max_a \mathbb{E}[R_h(s, a)]$: this is a natural choice for such states.
  Since $V^B = A^{-\ell} < V^D = A^{-\ell/2}$, the tree has Bernoulli leaves, the line has deterministic rewards, and both first-step rewards are zero. MPC estimates the tree's value at s_1 as $\widehat{V}_{\mathrm{Tr}} = V^B = A^{-\ell}$ and takes the line whose value is $\widehat{V}_L = A^{-\ell/2}$. The actual values of the paths are exactly

$$V_{\mathrm{Tr}} = V_{\mathrm{opt}} = 1 - (1 - A^{-\ell})^{A^\ell} \qquad \text{and} \qquad V_{\mathrm{L}} = V_{\mathrm{MPC}} = A^{-\ell/2};$$

  The displayed values are rounded. The values for $A \in \{2, 5\}$ and $\ell \in \{2, 5, 10\}$ are presented in Table 2.

Table 1: Exact-value illustration of Claim 2.1.

A	ℓ	S	$B^\star$	V_{FB}	V_{opt}	$V_{\mathrm{FB}}/V_{\mathrm{opt}}$
2	2	5	2	0.4375	0.6836	0.64
2	5	33	3	0.2243	0.6379	0.3516
2	10	1,025	6	0.03078	0.6323	0.04868
5	2	8	2	0.1846	0.6396	0.2887
5	5	783	3	0.03922	0.6322	0.06203
5	10	2,441,408	6	$3.199 \cdot 10^{-4}$	0.6321	5.062×10^{-4}

Table 2: Exact-value illustration of Claim I.1.

A	ℓ	S	$\widehat{V}_{\mathrm{Tr}}$	$\widehat{V}_{\mathrm{L}} = V_{\mathrm{L}} = V_{\mathrm{MPC}}$	$V_{\mathrm{Tr}} = V_{\mathrm{opt}}$	$V_{\mathrm{MPC}}/V_{\mathrm{opt}}$
2	2	7	0.25	0.5	0.6836	0.7314
2	5	38	0.03125	0.1768	0.6379	0.2771
2	10	1,035	9.766×10^{-4}	0.03125	0.6323	0.04942
5	2	10	0.04	0.2	0.6396	0.3127
5	5	788	0.00032	0.01789	0.6322	0.0283
5	10	2,441,418	1.024×10^{-7}	0.00032	0.6321	5.062×10^{-4}

J Table of Notations

Notation	Meaning
$\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{R}, P, H)$	Episodic tabular MDP.
$\mathcal{S}, \mathcal{A}$	State space and action space.
S, A	Cardinalities of the state and action spaces, respectively.
$\mathcal{R}_h$	Reward distribution at step h. The reward distribution of a state-action is $\mathcal{R}_h(s,a)$
P_h	Transition kernel at step h. The probability of moving to state s' from (s,a) is $P_h(s'\|s,a)$.
H	Episode horizon.
R_h	Realized reward at step h, with $R_h \sim \mathcal{R}_h(s_h, a_h)$.
s_h, a_h	State and action at step h.
π	Global policy used for the full interaction.
Π_{det}	Set of deterministic Markov policies.
ℓ	Maximum lookahead horizon.
$\ell_h = \min\{\ell, H-h+1\}$	Effective lookahead horizon at step h near the end of the episode.
ϕ	Local deterministic policy executed inside a lookahead batch.
$\tau^{\phi}_{h:t}(s)$ (1)	Realized trajectory from step h to step t starting at $s_h = s$ under local policy ϕ.
$I_{h,\ell}(s)$	Realized ℓ-step lookahead information at step h and state s; consists of all realized trajectories for all deterministic policies.
$\mathcal{I}_{h,\ell}(s)$	Distribution of lookahead information sampled at step h and state s.
$\mathfrak{R}_{t\|h}(s,\phi,I)$	Realized reward at time t induced by starting from (h,s), following ϕ, under lookahead realization I.
$\mathfrak{s}_{t\|h}(s,\phi,I)$	Realized state at time t induced by starting from (h,s), following ϕ, under lookahead realization I.
$B^{\pi}_h(s)$	Batch length chosen by an ABP π at step h in state s.
$V^{\pi}_h(s)$	The value of an ABP π upon starting a new batch at step h and state s.
$\mathfrak{t}^{\pi}_i$	The (random) step in which a policy π started its i^{th} batch.
E^n_h	The event in which some batch started at step h.
$\mathfrak{Q}^*_h(s,B;I,V)$ (2)	Optimal cumulative reward over a batch of length B starting from (h,s) under lookahead I, plus terminal value V at the end of the batch.
$B^*_h(s)$ (4)	Optimal batch length at step h in state s.
$V^*_h(s)$ (3)	Optimal value among ABPs from a new batch at step h and state s.
$Q^*_h(s,B)$	Expected value of starting a fresh batch of length B at (h,s) and continuing optimally afterward: $Q^*_h(s,B) = \mathbb{E}_{I\sim\mathcal{I}_{h,B}(s)}[\mathfrak{Q}^*_h(s,B;I,V^*_{h+B})]$.
$\phi^{*,h}$ (5)	Optimal local policy executed inside the batch started at step h.
$\mathfrak{J}^I_{h,B}(s,s')$ (6)	Maximum cumulative reward over B steps, under lookahead I, among local policies that start at step h in state s, and end in state s' at step $h+B$.
$\mathfrak{S}^I_{h,B}(s)$ (7)	Set of states reachable after B steps from (h,s) under lookahead realization I.
$\boldsymbol{R}, \boldsymbol{s'}$	Lookahead information in a table format (as described in Appendix A).
$\mathcal{M}^{\mathrm{B}}$	Augmented MDP used in the analysis/planning reduction for ABPs (see Appendix B).
$V^{\pi}_t(s, \boldsymbol{R}, \boldsymbol{s'}, h, B\|\mathcal{M}^{\mathrm{B}})$	Value in the augmented MDP, with $V^{\pi}_t(s,h\|\mathcal{M}^{\mathrm{B}}) = V^{\pi}_t(s, 0^{\otimes SA\ell}, \emptyset^{\otimes SA\ell}, h, 0\|\mathcal{M}^{\mathrm{B}})$.
$\mathrm{Reg}(K)$	Regret after K episodes: $\sum_{k=1}^{K}\big(V^*_1(s^k_1) - V^{\pi^k}_1(s^k_1)\big)$.
K	Number of episodes.
π^k	Policy executed in episode k.
s^k_1	Initial state of episode k.
$b^k_h(s,B)$ (18)	Exploration bonus used by the learning algorithm.
$n^k_h(s,B)$	Number of observations collected before episode k for step h, state s, and batch length B.
$\widehat{\mathrm{Var}}^k_h(s,B;V)$ (10)	Empirical variance Q-value estimate assuming terminal value V.
$\bar{V}^k_h$	Optimistic value estimate at episode k.
L^k_δ	Logarithmic confidence term: $L^k_\delta = \ln\Big(\frac{18SH\ell k^3(k+1)}{\delta}\Big)$.
δ	Confidence parameter.
$\mathrm{Var}^*_h(s,B)$ (19)	Variance of the optimal batch return when starting a fresh batch of length B at (h,s).
$\mathfrak{V}^{\pi}_h(s,I)$ (20)	Return of ABP π when starting a fresh batch at (h,s) under lookahead realization I.
$\mathrm{Var}^{\pi}_h(s)$ (21)	Variance of the random batch return $\mathfrak{V}^{\pi}_h(s,I)$.
$E^{opt}(k), E^{\widehat{\mathrm{Var}}}(k), E^{Err}(k)$	Good events defined in Appendix D.1.
$E^{\mathrm{diff}}, E^{\mathrm{Var}}$	Good events defined in Appendix D.3.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: In the paper, we describe the lookahead model in detail, show failure modes of existing common approaches and present in detail the policy space we study – ABPs – including planning and learning results.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: All assumptions are clearly stated in the paper and future directions to mitigate them are discussed in the future work section.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [Yes]

Justification: All assumptions are provided in the paper, except for Claim 2.2, for which an informal version of the assumption is given in the main paper and the formal assumptions are stated in the appendix. All proofs can be found in the appendix.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [N/A]

   Justification: The paper is theoretical in nature and does not include experiments.

   Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper does not include experiments.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [N/A]

Justification: The paper does not include experiments.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [N/A]

Justification: The paper does not include experiments.

Guidelines:

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

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

8. **Experiments compute resources**

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

   Answer: [N/A]

   Justification: The paper does not include experiments.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: As a theoretical work, the research conducted in the paper conform with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [N/A]

    Justification: This paper theoretically studies the problem of planning and learning with multi-step lookahead information. While the applications for this setting vary, and some applications might have some societal consequences, our paper is theoretical in nature, and we do not feel that it has any direct societal impact that should be specifically highlighted.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: As a theoretical work, the paper poses no such risk.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [N/A]

    Justification: The paper does not use existing assets.

    Guidelines:

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

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

13. **New assets**

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

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

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

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

    Answer: [N/A]

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

    Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: LLMs were only used for formatting, proofreading, and diagram generation.

Guidelines:

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