Open Loop Optimistic Planning

Sébastien Bubeck, Rémi Munos
SequeL Project, INRIA Lille
40 avenue Halley,
59650 Villeneuve d'Ascq, France
{sebastien.bubeck, remi.munos}@inria.fr

Abstract

We consider the problem of planning in a stochastic and discounted environment with a limited numerical budget. More precisely, we investigate strategies exploring the set of possible sequences of actions, so that, once all available numerical resources (e.g. CPU time, number of calls to a generative model) have been used, one returns a recommendation on the best possible immediate action to follow based on this exploration. The performance of a strategy is assessed in terms of its simple regret, that is the loss in performance resulting from choosing the recommended action instead of an optimal one. We first provide a minimax lower bound for this problem, and show that a uniform planning strategy matches this minimax rate (up to a logarithmic factor). Then we propose a UCB (Upper Confidence Bounds)-based planning algorithm, called OLOP (Open-Loop Optimistic Planning), which is also minimax optimal, and prove that it enjoys much faster rates when there is a small proportion of near-optimal sequences of actions. Finally, we compare our results with the regret bounds one can derive for our setting with bandits algorithms designed for an infinite number of arms.

1 Introduction

We consider the problem of planning in general stochastic and discounted environments. More precisely, the decision making problem consists in an exploration phase followed by a recommendation. First, the agent explores freely the set of possible sequences of actions (taken from a finite set A of cardinality K), using a finite budget of n actions. Then the agent makes a recommendation on the first action $a(n) \in A$ to play. This decision making problem is described precisely in Figure 1. The goal of the agent is to find the best way to explore its environment (first phase) so that, once the available resources have been used, he is able to make the best possible recommendation on the action to play in the environment.

During the exploration of the environment, the agent iteratively selects sequences of actions, under the global constraint that he can not take more than n actions in total, and receives a reward after each action. More precisely, at time step t during the m^{th} sequence, the agent played $a^m_{1:t} = a^m_1 \dots a^m_t \in A^t = A \times \dots A$ and receives a discounted reward $\gamma^t Y^m_t$ where $\gamma \in (0,1)$ is the discount factor. We make a stochastic assumption on the generating process for the reward: given $a^m_{1:t}$, Y_t is drawn from a probability distribution $\nu(a^m_{1:t})$ on $[0,1]$. Given $a \in A^t$, we write $\mu(a)$ for the mean of the probability $\nu(a)$.

The performance of the recommended action $a(n) \in A$ is assessed in terms of the so-called **simple regret** r_n, which is the performance loss resulting from choosing this sequence and then following an optimal path instead of following an optimal path from the beginning:

$$r_n = V - V(a(n)),$$

where $V(a(n))$ is the (discounted) value of the action (or sequence) $a(n)$, defined for any finite sequence of actions $a \in A^h$ as:

$$V(a) = \sup_{u \in A^\infty : u_{1:h} = a} \sum_{t \geq 1} \gamma^t \mu(u_{1:t}), \tag{1}$$

and V is the optimal value, that is the maximum expected sum of discounted rewards one may obtain (i.e. the sup in (1) is taken over all sequences in A^∞).

Note that this simple regret criterion has already been studied in multi-armed bandit problems, see Bubeck et al. (2009a); Audibert et al. (2010).

Exploration in a stochastic and discounted environment.

Parameters available to the agent: discount factor $\gamma \in (0,1)$, number of actions K, number of rounds n.

Parameters unknown to the agent: the reward distributions $\nu(a)$, $a \in A^*$.

For each episode $m \geq 1$; for each moment in the episode $t \geq 1$;

(1) If n actions have already been performed then the agent outputs an action $a(n) \in A$ and the game stops.

(2) The agent chooses an action $a_t^m \in A$.

(3) The environment draws $Y_t^m \sim \nu(a_{1:t}^m)$ and the agent receives the reward $\gamma^t Y_t^m$.

(4) The agent decides to either move the next moment $t+1$ in the episode or to reset to its initial position and move the next episode $m+1$.

Goal: maximize the value of the recommended action (or sequence): $V(a(n))$ (see (1) for the definition of the value of an action).

Figure 1: Exploration in a stochastic and discounted environment.

An important application of this framework concerns the problem of planning in Markov Decision Processes (MDPs) with very large state spaces. We assume that the agent possesses a generative model which enables to generate a reward and a transition from any state-action to a next state, according to the underlying reward and transition model of the MDP. In this context, we propose to use the generative model to perform a planning from the current state (using a finite budget of n calls to a generative model) to generate a near-optimal action $a(n)$ and then apply $a(n)$ in the real environment. This action modifies the environment and the planning procedure is repeated from the next state to select the next action and so on. From each state, the planning consists in the exploration of the set of possible sequences of actions as described in Figure 1, where the generative model is used to generate the rewards.

Note that, using control terminology, the setting described above (from a given state) is called "open-loop" planning, because the class of considered policies (i.e. sequences of actions) are only function of time (and not of the underlying resulting states). This open-loop planning is in general sub-optimal compared to the optimal (closed-loop) policy (mapping from states to actions). However, here, while the planning is open-loop (i.e. we do not take into consideration the subsequent states in the planning), the resulting general policy is closed-loop (since the chosen action depends on the current state).

This approach to MDPs has already been investigated as an alternative to usual dynamic programming approaches (which approximate the optimal value function to design a near optimal policy) to circumvent the computational complexity issues. For example, Kearns et al. describe a sparse sampling method that uses a finite amount of computational resources to build a look-ahead tree from the current state, and returns a near-optimal action with high probability.

Another field of application is POMDPs (Partially Observable Markov Decision Problems), where from the current belief state an open-loop plan may be built to select a near-optimal immediate action (see e.g. Yu et al. (2005); Hsu et al. (2007)). Note that, in these problems, it is very common to have a limited budget of computational resources (CPU time, memory, number of calls to the generative model, ...) to select the action to perform in the real environment, and we aim at making an efficient use of the available resources to perform the open-loop planning.

Moreover, in many situations, the generation of state-transitions is computationally expensive, thus it is critical to make the best possible use of the available number of calls to the model to output the action. For instance, an important problem in waste-water treatment concerns the control of a biochemical process for anaerobic digestion. The chemical reactions involve hundreds of different bacteria and the simplest models of the dynamics already involve dozens of variables (for example, the well-known model called ADM1 Batstone et al. (2002) contains 32 state variables) and their simulation is numerically heavy. Because of the curse of dimensionality, it is impossible to compute an optimal policy for such model. The methodology described above aims at a less ambitious goal, and search for a closed-loop policy which is open-loop optimal at each time step. While this policy is suboptimal, it is also a more reasonable target in terms of computational complexity. The strategy considered here proposes to use the model to simulate transitions and perform a complete open-loop planning at each time step.

The main contribution of the paper is the analysis of an adaptive exploration strategy of the search space, called Open-Loop Optimistic Planning (OLOP), which is based on the "optimism in the face of uncertainty" principle, i.e. where the most promising sequences of actions are explored first. The idea of optimistic planning has already been investigated in the simple case of deterministic environments, Hren and Munos (2008). Here we consider the non-trivial extension of this optimistic approach to planning in stochastic environments. For that purpose, upper confidence bounds (UCBs) are assigned to all sequences of actions, and the exploration expands further the sequences with highest UCB. The idea of selecting actions based on UCBs comes from the multi-armed bandits literature, see Lai and Robbins (1985); Auer et al. (2002). Planning under uncertainty using UCBs has been considered previously in Chang et al. (2007) (the so-called UCB sampling) and in Kocsis and Szepesvari (2006), where the resulting algorithm, UCT (UCB applied to Trees), has been successfully applied to the large scale tree search problem of computer-go, see Gelly et al. (2006). However, its regret analysis shows that UCT may perform very poorly because of overly-optimistic assumptions in the design of the bounds, see Coquelin and Munos (2007). Our work is close in spirit to BAST (Bandit Algorithm for Smooth Trees), Coquelin and Munos (2007), the Zooming Algorithm, Kleinberg et al. (2008) and HOO (Hierarchical Optimistic Optimization), Bubeck et al. (2009b). Like in these previous works, the performance bounds of OLOP are expressed in terms of a measure of the proportion of near-optimal paths.

However, as we shall discuss in Section 4, these previous algorithms fail to obtain minimax guarantees for our problem. Indeed, a particularity of our planning problem is that the value of a sequence of action is defined as the sum of discounted rewards along the path, thus the rewards obtained along any sequence provides information, not only about that specific sequence, but also about any other sequence sharing the same initial actions. OLOP is designed to use this property as efficiently as possible, to derive tight upper-bounds on the value of each sequence of actions.

Note that our results does not compare with traditional regret bounds for MDPs, such as the ones proposed in Auer et al. (2009). Indeed, in this case one compares to the optimal closed-loop policy, and the resulting regret usually depends on the size of the state space (as well as on other parameters of the MDP).

Outline. We exhibit in Section 2 the minimax rate (up to a logarithmic factor) for the simple regret in discounted and stochastic environments: both lower and upper bounds are provided. Then in Section 3 we describe the OLOP strategy, and show that if there is a small proportion of near-optimal sequences of actions, then faster rates than minimax can be derived. In Section 4 we compare our results with previous works and present several open questions. Finally the Appendix contains the analysis of OLOP.

Notations To shorten the equations we use several standard notations over alphabets. We collect them here: $A^0 = \{\emptyset\}$, A^* is the set of finite words over A (including the null word $\emptyset$), for $a \in A^*$ we note $h(a)$ the integer such that $a \in A^{h(a)}$, $aA^h = \{ab, b \in A^h\}$, for $a \in A^h$ and $h' > h$ we note $a_{1:h'} = a\emptyset \dots \emptyset$ and $a_{1:0} = \emptyset$.

2 Minimax optimality

In this section we derive a lower bound on the simple regret (in the worst case) of any agent, and propose a simple (uniform) forecaster which attains this optimal minimax rate (up to a logarithmic factor). The main purpose of the section on the uniform planning is to show explicitly the special concentrations property that our model enjoys.

2.1 Minimax lower bound

We propose here a new lower bound, whose proof follows from a simple adaptation of the technique developed in Auer et al. (2003). Note that this lower bound is not a particular case of the ones derived in Kleinberg et al. (2008) or Bubeck et al. (2009b) in a more general framework, as we shall see in Section 4.

Theorem 1 *Any agent satisfies:*

$$\sup_{\nu} \mathbb{E} r_n = \begin{cases} \Omega\left(\left(\frac{\log n}{n}\right)^{\frac{\log 1/\gamma}{\log K}} \right) & \text{if } \gamma\sqrt{K} > 1, \\ \Omega\left(\sqrt{\frac{\log n}{n}} \right) & \text{if } \gamma\sqrt{K} \leq 1. \end{cases}$$

2.2 Uniform Planning

To start gently, let us consider first (and informally) a *naive version* of the uniform planning. One can choose a depth H, uniformly test all sequences of actions in A^H (with $(n/H)/K^H$ samples for each sequence), and

then return the empirical best sequence. Cutting the sequences at depth H implies an error of order γ^H, and relying on empirical estimates with $(n/H)/K^H$ samples adds an error of order $\sqrt{\frac{HK^H}{n}}$, leading to a simple regret bounded as $O\left(\gamma^H + \sqrt{\frac{HK^H}{n}}\right)$. Optimizing over H yields an upper bound on the simple regret of the naive uniform planning of order:

$$O\left(\left(\frac{\log n}{n}\right)^{\frac{\log 1/\gamma}{\log K + 2\log 1/\gamma}}\right), \tag{2}$$

which does not match the lower bound. The cautious reader probably understands why this version of uniform planning is suboptimal. Indeed we do not use the fact that any sequence of actions of the form ab gives information on the sequences ac. Hence, the concentration of the empirical mean for short sequences of actions is much faster than for long sequences. This is the critical property which enables us to fasten the rates with respect to traditional methods, see Section 4 for more discussion on this.

We describe now the *good version* of uniform planning. Let $H \in \mathbb{N}$ be the largest integer such that $HK^H \leq n$. Then the procedure goes as follows: For each sequence of actions $a \in A^H$, the uniform planning allocates one episode (of length H) to estimate the value of the sequence a, that is it receives $Y_t^a \sim \nu(a_{1:t})$, $1 \leq t \leq H$ (drawn independently). At the end of the allocation procedure, it computes for all $a \in A^h$, $h \leq H$, the empirical average reward of the sequence a:

$$\widehat{\mu}(a) = \frac{1}{K^{H-h}} \sum_{b \in A^H : b_{1:h} = a} Y_h^b.$$

(obtained with K^{H-h} samples.) Then, for all $a \in A^H$, it computes the empirical value of the sequence a:

$$\widehat{V}(a) = \sum_{t=1}^{H} \gamma^t \widehat{\mu}(a_{1:t}).$$

It outputs $a(n) \in A$ defined as the first action of the sequence $\arg\max_{a \in A^H} \widehat{V}(a)$ (ties break arbitrarily).

This version of uniform planning makes a much better use of the reward samples than the naive version. Indeed, for any sequence $a \in A^h$, it collects the rewards Y_h^b received for sequences $b \in aA^{H-h}$ to estimate $\mu(a)$. Since $|aA^{H-h}| = K^{H-h}$, we obtain an estimation error for $\mu(a)$ of order $\sqrt{K^{h-H}}$. Then, thanks to the discounting, the estimation error for $V(a)$, with $a \in A^H$, is of order $K^{-H/2} \sum_{h=1}^{H} (\gamma\sqrt{K})^h$. On the other hand, the approximation error for cutting the sequences at depth H is still of order γ^H. Thus, since H is the largest depth (given n and K) at which we can explore once each node, we obtain the following behavior: When K is large, precisely $\gamma\sqrt{K} > 1$, then H is small and the estimation error is of order γ^H, resulting in a simple regret of order $n^{-(\log 1/\gamma)/\log K}$. On the other hand, if γ is small, precisely $\gamma\sqrt{K} < 1$, then the depth H becomes less important, and the estimation error is of order $K^{-H/2}$, resulting in a simple regret of order $n^{-1/2}$. This reasoning can easily be made precise to prove the following Theorem.

Theorem 2 *The (good) uniform planning satisfies:*

$$\mathbb{E} r_n \leq \begin{cases} O\left(\sqrt{\log n}\left(\frac{\log n}{n}\right)^{\frac{\log 1/\gamma}{\log K}}\right) & \text{if } \gamma\sqrt{K} > 1, \\ O\left(\frac{(\log n)^2}{\sqrt{n}}\right) & \text{if } \gamma\sqrt{K} = 1, \\ O\left(\frac{\log n}{\sqrt{n}}\right) & \text{if } \gamma\sqrt{K} < 1. \end{cases}$$

Remark 1 *We do not know whether the $\sqrt{\log n}$ (respectively $(\log n)^{3/2}$ in the case $\gamma\sqrt{K} = 1$) gap between the upper and lower bound comes from a suboptimal analysis (either in the upper or lower bound) or from a suboptimal behavior of the uniform forecaster.*

3 OLOP (Open Loop Optimistic Planning)

The uniform planning described in Section 2.2 is a static strategy, it does not adapt to the rewards received in order to improve its exploration. A stronger strategy could select, at each round, the next sequence to explore as a function of the previously observed rewards. In particular, since the value of a sequence is the sum

of discounted rewards, one would like to explore more intensively the sequences starting with actions that already yielded high rewards. In this section we describe an adaptive exploration strategy, called Open Loop Optimistic Planning (OLOP), which explores first the most promising sequences, resulting in much stronger guarantees than the one derived for uniform planning.

OLOP proceeds as follows. It assigns upper confidence bounds (UCBs), called B-values, to all sequences of actions, and selects at each round a sequence with highest B-value. This idea of a UCB-based exploration comes from the multi-armed bandits literature, see Auer et al. (2002). It has already been extended to hierarchical bandits, Chang et al. (2007); Kocsis and Szepesvari (2006); Coquelin and Munos (2007), and to bandits in metric (or even more general) spaces, Auer et al. (2007); Kleinberg et al. (2008); Bubeck et al. (2009b).

Like in these previous works, we express the performance of OLOP in terms of a measure of the proportion of near-optimal paths. More precisely, we define $\kappa_c \in [1, K]$ as the branching factor of the set of sequences in A^h that are $c\frac{\gamma^{h+1}}{1-\gamma}$-optimal, where c is a positive constant, i.e.

$$\kappa_c = \limsup_{h\to\infty} \left| \left\{ a \in A^h : V(a) \geq V - c\frac{\gamma^{h+1}}{1-\gamma} \right\} \right|^{1/h}. \tag{3}$$

Intuitively, the set of sequences $a \in A^h$ that are $\frac{\gamma^{h+1}}{1-\gamma}$-optimal are the sequences for which the perfect knowledge of the discounted sum of mean rewards $\sum_{t=1}^h \gamma^t \mu(a_{1:t})$ is not sufficient to decide whether a belongs to an optimal path or not, because of the unknown future rewards for $t > h$. In the main result, we consider κ_2 (rather than κ_1) to account for an additional uncertainty due to the empirical estimation of $\sum_{t=1}^h \gamma^t \mu(a_{1:t})$. In Section 4, we discuss the link between κ and the other measures of the set of near-optimal states introduced in the previously mentioned works.

3.1 The OLOP algorithm

The OLOP algorithm is described in Figure 2. It makes use of some B-values assigned to any sequence of actions in A^L. At time $m = 0$, the B-values are initialized to $+\infty$. Then, after episode $m \geq 1$, the B-values are defined as follows: For any $1 \leq h \leq L$, for any $a \in A^h$, let

$$T_a(m) = \sum_{s=1}^m \mathbb{1}\{a^s_{1:h} = a\}$$

be the number of times we played a sequence of actions beginning with a. Now we define the empirical average of the rewards for the sequence a as:

$$\widehat{\mu}_a(m) = \frac{1}{T_a(m)} \sum_{s=1}^m Y^s_h \mathbb{1}\{a^s_{1:h} = a\},$$

if $T_a(m) > 0$, and 0 otherwise. The corresponding upper confidence bound on the value of the sequence of actions a is by definition:

$$U_a(m) = \sum_{t=1}^h \left(\gamma^t \widehat{\mu}_{a_{1:t}}(m) + \gamma^t \sqrt{\frac{2\log M}{T_{a_{1:t}}(m)}} \right) + \frac{\gamma^{h+1}}{1-\gamma},$$

if $T_a(m) > 0$ and $+\infty$ otherwise. Now that we have upper confidence bounds on the value of many sequences of actions we can sharpen these bounds for the sequences $a \in A^L$ by defining the B-values as:

$$B_a(m) = \inf_{1\leq h\leq L} U_{a_{1:h}}(m).$$

At each episode $m = 1, 2, \ldots, M$, OLOP selects a sequence $a^m \in A^L$ with highest B–value, observes the rewards $Y^m_t \sim \nu(a^m_{1:t})$, $t = 1, \ldots, L$ provided by the environment, and updates the B–values. At the end of the exploration phase, OLOP returns an action that has been the most played, *i.e.* $a(n) = \mathrm{argmax}_{a\in A} T_a(M)$.

3.2 Main result

Theorem 3 (Main Result) *Let $\kappa_2 \in [1, K]$ be defined by (3). Then, for any $\kappa' > \kappa_2$, OLOP satisfies:*

$$\mathbb{E}r_n = \begin{cases} \tilde{O}\left(n^{-\frac{\log 1/\gamma}{\log \kappa'}}\right) & \text{if } \gamma\sqrt{\kappa'} > 1, \\ \tilde{O}\left(n^{-\frac{1}{2}}\right) & \text{if } \gamma\sqrt{\kappa'} \leq 1. \end{cases}$$

(We say that $u_n = \tilde{O}(v_n)$ if there exists $\alpha, \beta > 0$ such that $u_n \leq \alpha(\log(v_n))^\beta v_n$)

Open Loop Optimistic Planning:

Let M be the largest integer such that $M\lceil \log M/(2\log 1/\gamma)\rceil \leq n$. Let $L = \lceil \log M/(2\log 1/\gamma)\rceil$.

For each episode $m = 1, 2, \ldots, M$;

(1) The agent computes the B–values at time $m-1$ for sequences of actions in A^L (see Section 3.1) and chooses a sequence that maximizes the corresponding B–value:

$$a^m \in \operatorname*{argmax}_{a \in A^L} B_a(m-1).$$

(2) The environment draws the sequence of rewards $Y_t^m \sim \nu(a_{1:t}^m)$, $t = 1, \ldots, L$.

Return an action that has been the most played: $a(n) = \operatorname{argmax}_{a \in A} T_a(M)$.

Figure 2: Open Loop Optimistic Planning

Remark 2 *One can see that the rate proposed for OLOP greatly improves over the uniform planning whenever there is a small proportion of near-optimal paths (i.e. κ is small). Note that this does not contradict the lower bound proposed in Theorem 1. Indeed κ provides a description of the environment ν, and the bounds are expressed in terms of that measure, one says that the bounds are distribution-dependent. Nonetheless, OLOP does not require the knowledge of κ, thus one can take the supremum over all $\kappa \in [1, K]$, and see that it simply replaces κ by K, proving that OLOP is minimax optimal (up to a logarithmic factor).*

Remark 3 *In the analysis of OLOP, we relate the simple regret to the more traditional* **cumulative regret***, defined at round n as $R_n = \sum_{m=1}^{M} \Big(V - V(a^m) \Big)$. Indeed, in the proof of Theorem 3, we first show that $r_n = \tilde{O}\left(\frac{R_n}{n}\right)$, and then we bound (in expectation) this last term. Thus the same bounds apply to $\mathbb{E}R_n$ with a multiplicative factor of order n. In this paper, we focus on the simple regret rather than on the traditional cumulative regret because we believe that it is a more natural performance criterion for the planning problem considered here. However note that OLOP is also minimax optimal (up to a logarithmic factor) for the cumulative regret, since one can also derive lower bounds for this performance criterion using the proof of Theorem 1.*

Remark 4 *One can also see that the analysis carries over to $r_n^L = V - V\left(\operatorname{argmax}_{a \in A^L} T_a(M)\right)$, that is we can bound the simple regret of a sequence of actions in A^L rather than only the first action $a(n) \in A$. Thus, using n actions for the exploration of the environment, one can derive a plan of length L (of order $\log n$) with the optimality guarantees of Theorem 3.*

4 Discussion

In this section we compare the performance of OLOP with previous algorithms that can be adapted to our framework. This discussion is summarized in Figure 3. We also point out several open questions raised by these comparisons.

Comparison with Zooming Algorithm/HOO: In Kleinberg et al. (2008) and Bubeck et al. (2009b), the authors consider a very general version of stochastic bandits, where the set of arms $\mathcal{X}$ is a metric space (or even more general spaces in Bubeck et al. (2009b)). When the underlying mean-payoff function is 1-Lipschitz with respect to the metric (again, weaker assumption are derived in Bubeck et al. (2009b)), the authors propose two algorithms, respectively the Zooming Algorithm and HOO, for which they derive performances in terms of either the zooming dimension or the near-optimality dimension. In a metric space, both of these notions coincide, and the corresponding dimension d is defined such that the number of balls of diameter ε required to cover the set of arms that are ε-optimal is of order ε^{-d}. Then, for both algorithms, one obtains a simple regret of order $\tilde{O}(n^{-1/(d+2)})$ (thanks to Remark 3).

Up to minor details, one can see our framework as a A^∞-armed bandit problem, where the mean-payoff function is the sum of discounted rewards. A natural metric ℓ on this space can be defined as follows: For any $a, b \in A^\infty$, $\ell(a,b) = \frac{\gamma^{h(a,b)+1}}{1-\gamma}$, where $h(a,b)$ is the maximum depth $t \geq 0$ such that $a_{1:t} = b_{1:t}$. One can very easily check that the sum of discounted reward is 1-Lipschitz with respect to that metric, since

$\sum_{t\geq 1}\gamma^t|\mu(a_{1:t})-\mu(b_{1:t})| = \sum_{t\geq h(a,b)+1}\gamma^t|\mu(a_{1:t})-\mu(b_{1:t})| \leq \ell(a,b)$. We show now that κ_2, defined by (3), is closely related to the near-optimality dimension. Indeed, note that the set aA^∞ can be seen as a ball of diameter $\frac{\gamma^{h(a)+1}}{1-\gamma}$. Thus, from the definition of κ_2, the number of balls of diameter $\frac{\gamma^{h+1}}{1-\gamma}$ required to cover the set of $2\frac{\gamma^{h+1}}{1-\gamma}$-optimal paths is of order of κ^h, which implies that the near-optimality dimension is $d = \frac{\log\kappa}{\log 1/\gamma}$. Thanks to this result, we can see that applying the Zooming Algorithm or HOO in our setting yield a simple regret bounded as:

$$\mathbb{E}r_n = \tilde{O}(n^{-1/(d+2)}) = \tilde{O}(n^{-\frac{\log 1/\gamma}{\log\kappa+2\log 1/\gamma}}). \tag{4}$$

Clearly, this rate is always worse than the ones in Theorem 3. In particular, when one takes the supremum over all κ, we find that (4) gives the same rate as the one of naive uniform planning in (2). This was expected since these algorithms do not use the specific shape of the global reward function (which is the sum of rewards obtained along a sequence) to generalize efficiently across arms. More precisely, they do not consider the fact that a reward sample observed for an arm (or sequence) ab provides strong information about any arm in aA^∞. Actually, the difference between HOO and OLOP is the same as the one between the naive uniform planning and the good one (see Section 2.2).

However, although things are obvious for the case of uniform planning, in the case of OLOP, it is much more subtle to prove that it is indeed possible to collect enough reward samples along sequences $ab, b \in A^*$ to deduce a sharp estimation of $\mu(a)$. Indeed, for uniform planning, if each sequence $ab, b \in A^h$ is chosen once, then one may estimate $\mu(a)$ using K^h reward samples. However in OLOP, since the exploration is expected to focus on promising sequences rather than being uniform, it is much harder to control the number of times a sequence $a \in A^*$ has been played. This difficulty makes the proof of Theorem 3 quite intricate compared to the proof of HOO for instance.

Comparison with UCB-AIR: When one knows that there are many near-optimal sequences of actions (i.e. when κ is close to K), then one may be convinced that among a certain number of paths chosen uniformly at random, there exists at least one which is very good with high probability. This idea is exploited by the UCB-AIR algorithm of Wang et al. (2009), designed for infinitely many-armed bandits, where at each round one chooses either to sample a new arm (or sequence in our case) uniformly at random, or to re-sample an arm that has already been explored (using a UCB-like algorithm to choose which one). The regret bound of Wang et al. (2009) is expressed in terms of the probability of selecting an ε-optimal sequence when one chooses the actions uniformly at random. More precisely, the characteristic quantity is β such that this probability is of order of ε^β. Again, one can see that κ_2 is closely related to β. Indeed, our assumption says that the proportion of ε-optimal sequences of actions (with $\varepsilon = 2\frac{\gamma^{h+1}}{1-\gamma}$) is $O(\kappa^h)$, resulting in $\kappa = K\gamma^\beta$. Thanks to this result, we can see that applying UCB-AIR in our setting yield a simple regret bounded as:

$$\mathbb{E}r_n = \begin{cases} \tilde{O}(n^{-\frac{1}{2}}) & \text{if } \kappa > K\gamma \\ \tilde{O}(n^{-\frac{1}{1+\beta}}) = \tilde{O}(n^{-\frac{\log 1/\gamma}{\log K/\kappa+\log 1/\gamma}}) & \text{if } \kappa \leq K\gamma \end{cases}$$

As expected, UCB-AIR is very efficient when there is a large proportion of near-optimal paths. Note that UCB-AIR requires the knowledge of β (or equivalently κ).

Figure 3 shows a comparison of the exponents in the simple regret bounds for OLOP, uniform planning, UCB-AIR, and Zooming/HOO (in the case $K\gamma^2 > 1$). We note that the rate for OLOP is better than UCB-AIR when there is a small proportion of near-optimal paths (small κ). Uniform planning is always dominated by OLOP and corresponds to a minimax lower bound for any algorithm. Zooming/HOO are always strictly dominated by OLOP and they do not attain minimax performances.

Comparison with deterministic setting: In Hren and Munos (2008), the authors consider a deterministic version of our framework, precisely they assume that the rewards are a deterministic function of the sequence of actions. Remarkably, in the case $\kappa\gamma^2 > 1$, we obtain the same rate for the simple regret as Hren and Munos (2008). Thus, in this case, we can say that planning in stochastic environments is not harder than planning in deterministic environments (moreover, note that in deterministic environments there is no distinction between open-loop and closed-loop planning).

Open questions: We identify four important open questions. (i) Is it possible to attain the performances of UCB-AIR when κ is unknown? (ii) Is it possible to improve OLOP if κ is known? (iii) Can we combine the advantages of OLOP and UCB-AIR to derive an exploration strategy with improved rate in intermediate cases (i.e. when $1/\gamma^2 < \kappa < \gamma K$)? (iv) What is a problem-dependent lower bound (in terms of κ or other measures of the environment) in this framework? Obviously these problems are closely related, and the current behavior of the bounds suggests that question (iv) might be tricky. As a side question, note that OLOP requires the knowledge of the time-horizon n, we do not know whether it is possible to obtain the same guarantees with an anytime algorithm.

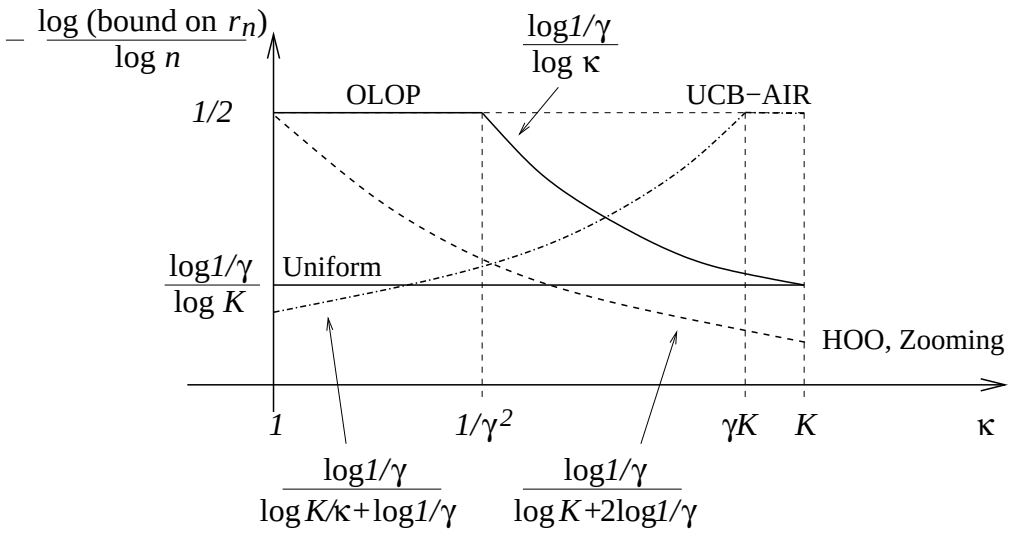

Figure 3: Comparison of the exponent rate of the bounds on the simple regret for OLOP, (good) uniform planning, UCB-AIR, and Zooming/HOO, as a function of $\kappa \in [1, K]$, in the case $K\gamma^2 > 1$.

References

J.-Y. Audibert, S. Bubeck, and R. Munos. Best arm identification in multi-armed bandits. In *23rd annual conference on learning theory*, 2010.

P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning Journal*, 47(2-3):235–256, 2002.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. Schapire. The non-stochastic multi-armed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2003.

P. Auer, R. Ortner, and C. Szepesvári. Improved rates for the stochastic continuum-armed bandit problem. *20th Conference on Learning Theory*, pages 454–468, 2007.

P. Auer, T. Jaksch, and R. Ortner. Near-optimal regret bounds for reinforcement learning. In Daphne Koller, Dale Schuurmans, Yoshua Bengio, and Lon Bottou, editors, *Advances in Neural Information Processing Systems 21*, page 89 96. MIT Press, 2009.

D. J. Batstone, J. Keller, I. Angelidaki, S. V. Kalyuzhnyi, S. G. Pavlostathis, A. Rozzi, W. T. M. Sanders, H. Siegrist, and V. A. Vavilin. Anaerobic digestion model no. 1 (adm1). *IWA Publishing*, 13, 2002.

S. Bubeck, R. Munos, and G. Stoltz. Pure exploration in multi-armed bandits problems. In *Proc. of the 20th International Conference on Algorithmic Learning Theory*, 2009a.

S. Bubeck, R. Munos, G. Stoltz, and Cs. Szepesvari. Online optimization in $\mathcal{X}$-armed bandits. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 201–208, 2009b.

Hyeong Soo Chang, Michael C. Fu, Jiaqiao Hu, and Steven I. Marcus. *Simulation-based Algorithms for Markov Decision Processes*. Springer, London, 2007.

P.-A. Coquelin and R. Munos. Bandit algorithms for tree search. In *Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence*, 2007.

J. L. Doob. *Stochastic Processes*. John Wiley & Sons, 1953.

S. Gelly, Y. Wang, R. Munos, and O. Teytaud. Modification of UCT with patterns in Monte-Carlo go. Technical Report RR-6062, INRIA, 2006.

W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13–30, 1963.

J.-F. Hren and R. Munos. Optimistic planning for deterministic systems. In *European Workshop on Reinforcement Learning*. 2008.

D. Hsu, W.S. Lee, and N. Rong. What makes some POMDP problems easy to approximate? In *Neural Information Processing Systems*, 2007.

M. Kearns, Y. Mansour, and A.Y. Ng. A sparse sampling algorithm for near-optimal planning in large Markovian decision processes. In *Machine Learning*, volume 49, year = 2002,, pages 193–208.

R. Kleinberg, A. Slivkins, and E. Upfal. Multi-armed bandits in metric spaces. In *Proceedings of the 40th ACM Symposium on Theory of Computing*, 2008.

L. Kocsis and Cs. Szepesvari. Bandit based Monte-carlo planning. In *Proceedings of the 15th European Conference on Machine Learning*, pages 282–293, 2006.

T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6:4–22, 1985.

Y. Wang, J.Y. Audibert, and R. Munos. Algorithms for infinitely many-armed bandits. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 1729–1736. 2009.

C. Yu, J. Chuang, B. Gerkey, G. Gordon, and A.Y. Ng. Open loop plans in POMDPs. Technical report, Stanford University CS Dept., 2005.

Appendix. Proof of Theorem 3

The proof of Theorem 3 is quite subtle. To present it in a gentle way we adopt a pyramidal proof rather than a pedagogic one. We propose seven lemmas, which we shall not motivate in depth, but prove in details. The precise architecture of the proof is as follows: Lemma 4 is a preliminary step, it justifies Remark 3. Then Lemma 5 underlines the important cases that we have to treat to show that suboptimal arms are not pulled too often. Lemma 6 takes care of one of these cases. Then, from Lemma 7 to 10, each Lemma builds on its predecessor. The main result eventually follows from Lemma 4 and 10 together with a simple optimization step.

We introduce first a few notations that will be useful. Let $1 \leq H \leq L$ and $a^* \in A^L$ such that $V(a^*) = V$. We define now some useful sets for any $1 \leq h \leq H$ and $0 \leq h' < h$;

$$\mathcal{I}_0 = \{\emptyset\}, \;\; \mathcal{I}_h = \left\{ a \in A^h : V - V(a) \leq \frac{2\gamma^{h+1}}{1-\gamma} \right\}, \;\; \mathcal{J}_h = \left\{ a \in A^h : a_{1:h-1} \in \mathcal{I}_{h-1} \;\text{ and }\; a \notin \mathcal{I}_h \right\}.$$

Note that, from the definition of κ_2, we have that for any $\kappa' > \kappa_2$, there exists a constant C such that for any $h \geq 1$,

$$|\mathcal{I}_h| \leq C\kappa'. \tag{5}$$

Now for $1 \leq m \leq M$, and $a \in A^t$ with $t \leq h$, write

$$\mathcal{P}^a_{h,h'}(m) = \left\{ b \in aA^{h-t} \cap \mathcal{J}_h : T_b(m) \geq \frac{8}{\gamma^2}(h+1)^2\gamma^{2(h'-h)} \log M + 1 \right\}.$$

Finally we also introduce the following random variable:

$$\tau^a_{h,h'}(m) = \mathbb{1}\left\{ T_a(m-1) < \frac{8}{\gamma^2}(h+1)^2\gamma^{2(h'-h)} \log M + 1 \leq T_a(m) \right\}.$$

Lemma 4 *The following holds true,*

$$r_n \leq \frac{2K\gamma^{H+1}}{1-\gamma} + \frac{3K}{M} \sum_{h=1}^{H} \sum_{a \in \mathcal{J}_h} \frac{\gamma^h}{1-\gamma} T_a(M).$$

Proof: Since $a(n) \in \arg\max_{a \in A} T_a(M)$, and $\sum_{a \in A} T_a(M) = M$, we have $T_{a(n)}(M) \geq M/K$, and thus:

$$\frac{M}{K}\Big(V - V(a(n))\Big) \leq \Big(V - V(a(n))\Big) T_{a(n)}(M) \leq \sum_{m=1}^{M} V - V(a^m).$$

Hence, we have, $r_n \leq \frac{K}{M} \sum_{m=1}^{M} V - V(a^m)$. Now remark that, for any sequence of actions $a \in A^L$, we have either:

- $a_{1:H} \in \mathcal{I}_H$; which implies $V - V(a) \leq \frac{2\gamma^{H+1}}{1-\gamma}$.

- or there exists $1 \leq h \leq H$ such that $a_{1:h} \in \mathcal{J}_h$; which implies $V - V(a) \leq V - V(a_{1:h-1}) + \frac{\gamma^h}{1-\gamma} \leq \frac{3\gamma^h}{1-\gamma}$.

Thus we can write:

$$\begin{aligned}\sum_{m=1}^{M} (V - V(a^m)) &= \sum_{m=1}^{M} (V - V(a^m)) \left(\mathbb{1}\{a^m \in \mathcal{I}_H\} + \mathbb{1}\{\exists 1 \leq h \leq H : a^m_{1:h} \in \mathcal{J}_h\} \right) \\ &\leq \frac{2\gamma^{H+1}}{1-\gamma} M + 3 \sum_{h=1}^{H} \sum_{a \in \mathcal{J}_h} \frac{\gamma^h}{1-\gamma} T_a(M),\end{aligned}$$

which ends the proof of Lemma 4. ■

The rest of the proof is devoted to the analysis of the term $\mathbb{E} \sum_{a \in \mathcal{J}_h} T_a(M)$. In the stochastic bandit literature, it is usual to bound the expected number of times a suboptimal action is pulled by the inverse sub-optimality (of this action) squared, see for instance Auer et al. (2002) or Bubeck et al. (2009b). Specialized to our setting, this implies a bound on $\mathbb{E}T_a(M)$, for $a \in \mathcal{J}_h$, of order γ^{-2h}. However, here, we obtain much stronger guarantees, resulting in the faster rates. Namely we show that $\mathbb{E} \sum_{a \in \mathcal{J}_h} T_a(M)$ is of order $(\kappa')^h$ (rather than $(\kappa')^h \gamma^{-2h}$ with previous methods).

The next lemma describes under which circumstances a suboptimal sequence of actions in $\mathcal{J}_h$ can be selected.

Lemma 5 *Let $0 \leq m \leq M-1$, $1 \leq h \leq L$ and $a \in \mathcal{J}_h$. If $a^{m+1} \in aA^*$ then it implies that one the three following propositions is true:*

$$\exists 1 \leq h' \leq L : U_{a^*_{1:h'}}(m) < V, \tag{6}$$

or

$$\sum_{t=1}^{h} \gamma^t \widehat{\mu}_{a_{1:t}}(m) \geq V(a) + \sum_{t=1}^{h} \gamma^t \sqrt{\frac{2 \log M}{T_{a_{1:t}}(m)}}, \tag{7}$$

or

$$2 \sum_{t=1}^{h} \gamma^t \sqrt{\frac{2 \log M}{T_{a_{1:t}}(m)}} > \frac{\gamma^{h+1}}{1-\gamma}. \tag{8}$$

Proof: If $a^{m+1} \in aA^*$ then it implies that $U_a(m) \geq \inf_{1 \leq h' \leq L} U_{a^*_{1:h'}}(m)$. That is either (6) is true or

$$U_a(m) = \sum_{t=1}^{h} \gamma^t \widehat{\mu}_{a_{1:t}}(m) + \gamma^t \sqrt{\frac{2 \log M}{T_{a_{1:t}}(m)}} + \frac{\gamma^{h+1}}{1-\gamma} \geq V.$$

In the latter case, if (7) is not satisfied, it implies

$$V(a) + 2 \sum_{t=1}^{h} \gamma^t \sqrt{\frac{2 \log M}{T_{a_{1:t}}(m)}} + \frac{\gamma^{h+1}}{1-\gamma} > V. \tag{9}$$

Since $a \in \mathcal{J}_h$ we have $V - V(a) - \frac{\gamma^{h+1}}{1-\gamma} \geq \frac{\gamma^{h+1}}{1-\gamma}$ which shows that equation (9) implies (8) and ends the proof. ■

We show now that both equations (6) and (7) have a vanishing probability of being satisfied.

Lemma 6 *The following holds true, for any $1 \leq h \leq L$ and $m \leq M$,*

$$\mathbb{P}(\textit{equation (6) or (7) is true}) \leq m(L+h)M^{-4} = \tilde{O}(M^{-3}).$$

Proof: Since $V \leq \sum_{t=1}^{h} \gamma^t \mu(a^*_{1:t}) + \frac{\gamma^{h+1}}{1-\gamma}$, we have,

$$\begin{aligned}
&\mathbb{P}(\exists\, 1 \leq h \leq L : U_{a^*_{1:h}}(m) \leq V) \\
&\leq \mathbb{P}\left(\exists\, 1 \leq h \leq L : \sum_{t=1}^{h} \gamma^t \left(\widehat{\mu}_{a^*_{1:t}}(m) + \sqrt{\frac{2\log M}{T_{a^*_{1:t}}(m)}}\right) \leq \sum_{t=1}^{h} \gamma^t \mu(a^*_{1:t}) \text{ and } T_{a^*_{1:h}}(m) \geq 1\right) \\
&\leq \mathbb{P}\left(\exists\, 1 \leq t \leq L : \widehat{\mu}_{a^*_{1:t}}(m) + \sqrt{\frac{2\log M}{T_{a^*_{1:t}}(m)}} \leq \mu(a^*_{1:t}) \text{ and } T_{a^*_{1:t}}(m) \geq 1\right) \\
&\leq \sum_{t=1}^{L} \mathbb{P}\left(\widehat{\mu}_{a^*_{1:t}}(m) + \sqrt{\frac{2\log M}{T_{a^*_{1:t}}(m)}} \leq \mu(a^*_{1:t}) \text{ and } T_{a^*_{1:t}}(m) \geq 1\right).
\end{aligned}$$

Now we want to apply a concentration inequality to bound this last term. To do it properly we exhibit a martingale and apply the Hoeffding-Azuma inequality for martingale differences (see Hoeffding (1963)). Let

$$S_j = \min\{s : T_{a^*_{1:t}}(s) = j\},\ j \geq 1.$$

If $S_j \leq M$, we define $\tilde{Y}_j = Y_t^{S_j}$, and otherwise $\tilde{Y}_j$ is an independent random variable with law $\nu(a^*_{1:t})$. We clearly have,

$$\begin{aligned}
&\mathbb{P}\left(\widehat{\mu}_{a^*_{1:t}}(m) + \sqrt{\frac{2\log M}{T_{a^*_{1:t}}(m)}} \leq \mu(a^*_{1:t}) \text{ and } T_{a^*_{1:t}}(m) \geq 1\right) \\
&= \mathbb{P}\left(\frac{1}{T_{a^*_{1:t}}(m)} \sum_{j=1}^{T_{a^*_{1:t}}(m)} \tilde{Y}_j + \sqrt{\frac{2\log M}{T_{a^*_{1:t}}(m)}} \leq \mu(a^*_{1:t}) \text{ and } T_{a^*_{1:t}}(m) \geq 1\right) \\
&\leq \sum_{u=1}^{m} \mathbb{P}\left(\frac{1}{u}\sum_{j=1}^{u} \tilde{Y}_j + \sqrt{\frac{2\log M}{u}} \leq \mu(a^*_{1:t})\right).
\end{aligned}$$

Now we have to prove that $\tilde{Y}_j - \mu(a^*_{1:t})$ is martingale differences sequence. This follows via an optional skipping argument, see (Doob, 1953, Chapter VII, Theorem 2.3). Thus we obtain

$$\mathbb{P}(\text{equation (6) is true}) \leq \sum_{t=1}^{L}\sum_{u=1}^{m} \exp\left(-2u\frac{2\log M}{u}\right) = LmM^{-4}.$$

The same reasoning gives

$$\mathbb{P}(\text{equation (7) is true}) \leq mhM^{-4},$$

which concludes the proof. ■

The next lemma proves that, if a sequence of actions has already been pulled enough, then equation (8) is not satisfied, and thus using lemmas 5 and 6 we deduce that with high probability this sequence of actions will not be selected anymore. This reasoning is made precise in Lemma 8.

Lemma 7 *Let $1 \leq h \leq L$, $a \in \mathcal{J}_h$ and $0 \leq h' < h$. Then equation (8) is not satisfied if the two following propositions are true:*

$$\forall 0 \leq t \leq h',\ T_{a_{1:t}}(m) \geq \frac{8}{\gamma^2}(h+1)^2\gamma^{2(t-h)}\log M, \tag{10}$$

and

$$T_a(m) \geq \frac{8}{\gamma^2}(h+1)^2\gamma^{2(h'-h)}\log M. \tag{11}$$

Proof: Assume that (10) and (11) are true. Then we clearly have:

$$\begin{aligned}
2\sum_{t=1}^{h}\gamma^t\sqrt{\frac{2\log M}{T_{a_{1:t}}(m)}} &= 2\mathbb{1}_{h'>0}\sum_{t=1}^{h'}\gamma^t\sqrt{\frac{2\log M}{T_{a_{1:t}}(m)}} + 2\sum_{t=h'+1}^{h}\gamma^t\sqrt{\frac{2\log M}{T_{a_{1:t}}(m)}} \\
&\leq \frac{\gamma^{h+1}}{h+1}h' + \frac{\gamma^{h+1}}{h+1}\sum_{t=h'+1}^{h}\gamma^{t-h'} \\
&\leq \frac{\gamma^{h+1}}{h+1}\left(h' + \frac{\gamma}{1-\gamma}\right) \\
&\leq \frac{\gamma^{h+1}}{1-\gamma},
\end{aligned}$$

which proves the result. ■

Lemma 8 *Let $1 \leq h \leq L$, $a \in \mathcal{J}_h$ and $0 \leq h' < h$. Then $\tau^a_{h,h'}(m+1) = 1$ implies that either equation (6) or (7) is satisfied or the following proposition is true:*

$$\exists 0 \leq t \leq h' : |\mathcal{P}^{a_{1:t}}_{h,h'}(m)| < \gamma^{2(t-h')}. \tag{12}$$

Proof: If $\tau^a_{h,h'}(m+1) = 1$ then it means that $a^{m+1} \in aA^*$ and (11) is satisfied. By Lemma 5 this implies that either (6), (7) or (8) is true and (11) is satisfied. Now by Lemma 7 this implies that (6) is true or (7) is true or (10) is false. We now prove that if (12) is not satisfied then (10) is true, which clearly ends the proof. This follows from: For any $0 \leq t \leq h'$,

$$T_{a_{1:t}}(m) = \sum_{b \in a_{1:t}A^{h-t}} T_b(m) \geq \gamma^{2(t-h')} \frac{8}{\gamma^2}(h+1)^2 \gamma^{2(h'-h)} \log M = \frac{8}{\gamma^2}(h+1)^2 \gamma^{2(t-h)} \log M.$$

■

The next lemma is the key step of our proof. Intuitively, using lemmas 5 and 8, we have a good control on sequences for which equation (12) is satisfied. Note that (12) is a property which depends on sub-sequences of a from length 1 to h'. In the following proof we will iteratively "drop" all sequences which do not satisfy (12) from length t onwards, starting from $t = 1$. Then, on the remaining sequences, we can apply Lemma 8.

Lemma 9 *Let $1 \leq h \leq L$ and $0 \leq h' < h$. Then the following holds true,*

$$\mathbb{E}|\mathcal{P}^{\emptyset}_{h,h'}(M)| = \tilde{O}\left(\gamma^{-2h'} \mathbb{1}_{h'>0} \sum_{t=0}^{h'} (\gamma^2\kappa')^t + (\kappa')^h M^{-2}\right).$$

Proof: Let $h' \geq 1$ and $0 \leq s \leq h'$. We introduce the following random variables:

$$m^a_s = \min\left(M, \min\left\{m \geq 0 : |\mathcal{P}^a_{h,h'}(m)| \geq \gamma^{2(s-h')}\right\}\right).$$

We will prove recursively that,

$$|\mathcal{P}^{\emptyset}_{h,h'}(m)| \leq \sum_{t=0}^{s} \gamma^{2(t-h')}|\mathcal{I}_t| + \sum_{a \in \mathcal{I}_s} \left|\mathcal{P}^a_{h,h'} \setminus \cup_{t=0}^{s} \mathcal{P}^{a_{1:t}}_{h,h'}(m^{a_{1:t}}_t)\right|. \tag{13}$$

The result is true for $s = 0$ since $\mathcal{I}_0 = \{\emptyset\}$ and by definition of $m^{\emptyset}_0$,

$$|\mathcal{P}^{\emptyset}_{h,h'}(m)| \leq \gamma^{-2h'} + |\mathcal{P}^{\emptyset}_{h,h'}(m) \setminus \mathcal{P}^{\emptyset}_{h,h'}(m^{\emptyset}_0)|.$$

Now let us assume that the result is true for $s < h'$. We have:

$$\begin{aligned}
\sum_{a \in \mathcal{I}_s} \left|\mathcal{P}^a_{h,h'}(m) \setminus \cup_{t=0}^{s} \mathcal{P}^{a_{1:t}}_{h,h'}(m^{a_{1:t}}_t)\right| &= \sum_{a \in \mathcal{I}_{s+1}} \left|\mathcal{P}^a_{h,h'}(m) \setminus \cup_{t=0}^{s} \mathcal{P}^{a_{1:t}}_{h,h'}(m^{a_{1:t}}_t)\right| \\
&\leq \sum_{a \in \mathcal{I}_{s+1}} \gamma^{2(s+1-h')} + \left|\mathcal{P}^a_{h,h'}(m) \setminus \cup_{t=0}^{s+1} \mathcal{P}^{a_{1:t}}_{h,h'}(m^{a_{1:t}}_t)\right| \\
&= \gamma^{2(s+1-h')}|\mathcal{I}_{s+1}| + \sum_{a \in \mathcal{I}_{s+1}} \left|\mathcal{P}^a_{h,h'}(m) \setminus \cup_{t=0}^{s+1} \mathcal{P}^{a_{1:t}}_{h,h'}(m^{a_{1:t}}_t)\right|,
\end{aligned}$$

which ends the proof of (13). Thus we proved (by taking $s = h'$ and $m = M$):

$$\begin{aligned}
|\mathcal{P}^{\emptyset}_{h,h'}(M)| &\leq \sum_{t=0}^{h'} \gamma^{2(t-h')}|\mathcal{I}_t| + \sum_{a \in \mathcal{I}_{h'}} \left|\mathcal{P}^a_{h,h'}(M) \setminus \cup_{t=0}^{s+1} \mathcal{P}^{a_{1:t}}_{h,h'}(m^{a_{1:t}}_t)\right| \\
&= \sum_{t=0}^{h'} \gamma^{2(t-h')}|\mathcal{I}_t| + \sum_{a \in \mathcal{J}_h} \left|\mathcal{P}^a_{h,h'}(M) \setminus \cup_{t=0}^{s+1} \mathcal{P}^{a_{1:t}}_{h,h'}(m^{a_{1:t}}_t)\right|
\end{aligned}$$

Now, for any $a \in \mathcal{J}_h$, let $\tilde{m} = \max_{0\le t\le h'} m_t^{a_{1:t}}$. Note that for $m \ge \tilde{m}$, equation (12) is not satisfied. Thus we have

$$\left|\mathcal{P}_{h,h'}^a \setminus \cup_{t=0}^{s+1}\mathcal{P}_{h,h'}^{a_{1:t}}(m_t^{a_{1:t}})\right| = \sum_{m=\tilde{m}}^{M-1}\tau_{h,h'}^a(m+1) = \sum_{m=0}^{M-1}\tau_{h,h'}^a(m+1)\,\mathbb{1}\{(12)\text{ is not satisfied}\}$$
$$\le \sum_{m=0}^{M-1}\tau_{h,h'}^a(m+1)\,\mathbb{1}\{(6)\text{ or }(7)\text{ is satisfied}\}.$$

where the last inequality results from Lemma 8. Hence, we proved:

$$|\mathcal{P}_{h,h'}^{\emptyset}| \le \sum_{t=0}^{h'}\gamma^{2(t-h')}|\mathcal{I}_t| + \sum_{m=0}^{M-1}\sum_{a\in\mathcal{J}_h}\mathbb{1}\{(6)\text{ or }(7)\text{ is satisfied}\}.$$

Taking the expectation, using (5) and applying Lemma 6 yield the claimed bound for $h' \ge 1$.

Now for $h' = 0$ we need a modified version of Lemma 8. Indeed in this case one can directly prove that $\tau_{h,0}^a(m+1) = 1$ implies that either equation (6) or (7) is satisfied (this follows from the fact that $\tau_{h,0}^a(m+1) = 1$ always imply that (10) is true for $h' = 0$). Thus we obtain:

$$|\mathcal{P}_{h,h'}^{\emptyset}| = \sum_{m=0}^{M-1}\sum_{a\in\mathcal{J}_h}\tau_{h,0}^a(m+1) \le \sum_{m=0}^{M-1}\sum_{a\in\mathcal{J}_h}\mathbb{1}\{(6)\text{ or }(7)\text{ is satisfied}\}.$$

Taking the expectation and applying Lemma 6 yield the claimed bound for $h' = 0$ and ends the proof. ■

Lemma 10 *Let $1 \le h \le L$. The following holds true,*

$$\mathbb{E}\sum_{a\in\mathcal{J}_h}T_a(M) = \tilde{O}\left(\gamma^{-2h}\sum_{h'=1}^{h}(\gamma^2\kappa')^{h'} + (\kappa')^h(1+\gamma^{-2h}M^{-2})\right).$$

Proof: We have the following computations:

$$\sum_{a\in\mathcal{J}_h}T_a(M) = \sum_{a\in\mathcal{J}_h\setminus\mathcal{P}_{h,h-1}^{\emptyset}}T_a(M) + \sum_{h'=1}^{h-1}\sum_{a\in\mathcal{P}_{h,h'}^{\emptyset}\setminus\mathcal{P}_{h,h'-1}^{\emptyset}}T_a(M) + \sum_{a\in\mathcal{P}_{h,0}^{\emptyset}}T_a(M)$$
$$\le \frac{8}{\gamma^2}(h+1)^2\gamma^{2(h-1-h)}|\mathcal{J}_h| + \sum_{h'=1}^{h-1}\frac{8}{\gamma^2}(h+1)^2\gamma^{2(h'-1-h)}\log M|\mathcal{P}_{h,h'}^{\emptyset}| + M|\mathcal{P}_{h,0}^{\emptyset}|$$
$$= \tilde{O}\left((\kappa')^h + \gamma^{-2h}\sum_{h'=1}^{h-1}\gamma^{2h'}|\mathcal{P}_{h,h'}^{\emptyset}| + M|\mathcal{P}_{h,0}^{\emptyset}|\right).$$

Taking the expectation and applying the bound of Lemma 9 gives the claimed bound. ■

Thus by combining Lemma 4 and 10 we obtain for $\kappa'\gamma^2 \le 1$:

$$\mathbb{E}r_n = \tilde{O}\left(\gamma^H + \gamma^{-H}M^{-1} + (\kappa')^H\gamma^{-H}M^{-3}\right),$$

and for $\kappa'\gamma^2 > 1$:

$$\mathbb{E}r_n = \tilde{O}\left(\gamma^H + (\kappa'\gamma)^H M^{-1} + (\kappa')^H\gamma^{-H}M^{-3}\right).$$

Thus in the case $\kappa'\gamma^2 \le 1$, taking $H = \lfloor \log M/(2\log 1/\gamma)\rfloor$ yields the claimed bound; while for $\kappa'\gamma^2 > 1$ we take $H = \lfloor \log M/\log\kappa'\rfloor$. Note that in both cases we have $H \le L$ (as it was required at the beginning of the analysis).