Scale-free adaptive planning for deterministic dynamics & discounted rewards

Peter L. Bartlett[1] **Victor Gabillon**[2] **Jennifer Healey**[3] **Michal Valko**[4]

Abstract

We address the problem of planning in an environment with deterministic dynamics and stochastic discounted rewards under a limited numerical budget where the ranges of both rewards and noise are unknown. We introduce PlaTγPOOS, an adaptive, robust, and efficient alternative to the OLOP (open-loop optimistic planning) algorithm. Whereas OLOP requires a priori knowledge of the ranges of both rewards and noise, PlaTγPOOS dynamically adapts its behavior to both. This allows PlaTγPOOS to be immune to two vulnerabilities of OLOP: failure when given underestimated ranges of noise and rewards and inefficiency when these are overestimated. PlaTγPOOS additionally adapts to the global smoothness of the value function. PlaTγPOOS acts in a provably more efficient manner vs. OLOP when OLOP is given an overestimated reward and show that in the case of no noise, PlaTγPOOS learns exponentially faster.

1. Introduction

We consider the problem of planning in a general *stochastic environment* with *deterministic dynamics* and *discounted rewards*. Our goal is to recommend the best first action for an agent to take from a given state. We envision that the discount factor γ is known and that our learner has a limited allocation of n interactions to spend querying a generative model of the environment. The objective is to maximize the sum of discounted rewards of the best sequence of actions following from the recommended first action. This is equivalent to minimizing the *simple regret*. We introduce the algorithm PlaTγPOOS, Planning wiTh γ Plus an Online Optimization Strategy, as a robust and efficient scale-free alternative to the OLOP algorithm (open-loop optimistic planning, Bubeck & Munos, 2010; Leurent & Maillard, 2019) for this setting. Our algorithm implements a scale-free function optimization strategy similar to SequOOL (Bartlett et al., 2019) rather than an upper-confidence-bound approach which allows our algorithm to efficiently adapt to the problem space *without prior knowledge of the ranges of the noise or the rewards*.

Planning in a stochastic environment is an important setting often modeled by Markov decision processes (MDPs, Puterman, 1994; Bertsekas & Tsitsiklis, 1996). One approach to solving these settings is to find the optimal policy that maximizes the expected sum of rewards and then generate an action recommendation according to that optimal policy. Unfortunately, in most practical settings where we are limited by computational resources, finding this optimal policy is often not possible, especially when the state space becomes large. Therefore, instead of trying to estimate the optimal policy of the MDP, we focus only on finding the *best first action* given our budget. We evaluate the performance of the recommendation in terms of the simple regret, the difference in reward between choosing the optimal first action vs. choosing our recommended first action and then in both cases choosing an optimal sequence of actions following the first action. This metric is often used to evaluate planning strategies that optimize numeric budgets (Bubeck & Munos, 2010; Buşoniu & Munos, 2012; Grill et al., 2016), in contrast to the cumulative regret where we are penalized during the search for querying sub-optimal actions. Once the agent takes the first action and moves to the next state, our evaluation can be repeated with a new budget allocation and the following best first action can be recommended. This allows us to approximately following an optimal policy, action by action, in an online way. Previously, there have been several strategies proposed on how to efficiently allocate a numeric budget to search for an optimal value in a stochastic space. Many of these have been successfully implemented using methods based on upper confidence bounds (UCBs) such as UCT (Upper Confidence Trees, Kocsis & Szepesvári, 2006). This approach has been proven to be very efficient in practice (Coulom, 2007; Gelly et al., 2006; Silver et al., 2016), however, UCT can badly misbehave on some problems (Coquelin & Munos, 2007) and more theoretically sound approaches have been proposed (Hren & Munos, 2008; Bubeck & Munos, 2010; Buşoniu & Munos, 2012; Feldman & Domshlak, 2014; Szörényi et al., 2014;

[1]University of California, Berkeley, USA [2]Noah's Ark Lab, Huawei Technologies, London, UK [3]Adobe Research, San Jose, USA [4]SequeL team, INRIA Lille - Nord Europe, France. Correspondence to: Victor Gabillon <victor.gabillon@huawei.com>.

Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

Kaufmann & Koolen, 2017; Shah et al., 2019). Some of these methods are connected to the ones from function optimization (Bubeck et al., 2011; Munos, 2011; Valko et al., 2013) as shown by Munos (2014), however, one key difference is that in planning, as opposed to function optimization, the structure of the reward is a discounted reward, specifically a *sum* of rewards and the *discount factor* γ. This reward structure influences the behavior of the optimizers (Bubeck & Munos, 2010), for example, the discount factor brings smoothness of the value function which in turn makes it easier to optimize. `PlaT`γ`POOS` exploits the effect of the discount factor to efficiently manage an adaptive planning strategy in the face of unknown ranges of noise and rewards.

This adaptive strategy of `PlaT`γ`POOS` can make it more robust and efficient in practice than other planning strategies. For example, even though they are theoretically sound, the empirical performance UCB-based approaches depend on the careful tuning of the upper confidence bound. If the upper confidence bound is too large then the UCB-based learner plays very conservatively by overestimating suboptimal options for many rounds. Moreover, these UCBs might depend on instance parameters that are simply not known such as the *range of the rewards* and the *range of the noise*. We build on the function optimization approach of Bartlett et al. (2019) that does not use UCBs and obtains improved results over the state-of-the-art by adapting to the problem difficulty with a scale-free approach. `PlaT`γ`POOS` adapts this scale-free optimization approach to planning. This *scale-free* property becomes a more desired feature as machine learning gets closer to applications, whether it is online (Ross et al., 2013; Orabona & Pál, 2018) or deep learning (Orabona & Tommasi, 2017), since in the reality, many parameters are never known.

In terms of planning strategy, the `PlaT`γ`POOS` algorithm is an adaptive, robust, and efficient alternative to `OLOP`. Whereas `OLOP` *requires the knowledge of where the ranges of both rewards and noise*, `PlaT`γ`POOS` dynamically adapts its behavior to the both ranges, as well as some potential additional global smoothness of the value function. Our algorithm's ability to adapt allows it to avoid failure in cases where the ranges of noise and rewards are underestimated and to act more efficiently in cases where they are overestimated. `PlaT`γ`POOS` recovers the results of `OLOP` while allowing improvements in various classes of problems.

Our contributions We show that `PlaT`γ`POOS`:

- adapts its behavior to an unknown range of rewards,
- requires no apriori assumptions or knowledge on noise,
- empirically learns much faster than UCB approaches,
- gets the fast rate of deterministic planning in low noise for all regime; in particular, it learns exponentially faster than `OLOP` when there happens to be no noise,
- adapts also the global smoothness ρ and ν beyond the base smoothness provided by γ.

We additionally address a realistic constraint where the agent can only *reset* to the original state and not to any state it wishes. Our results hold for MDPs with deterministic dynamics and can equally be applied to open loop planning problems (as discussed by Munos 2014) where we search for the best sequence of actions, ignoring the actual states that are reached after each action (Bubeck & Munos, 2010).

Related algorithms, where the objective is to find the value of the state rather than to identify the best action, include `TrailBlazer` (Grill et al., 2016) and `StOP` (Szörényi et al., 2014). A key difference is that these algorithms are *fixed confidence* and output a value using a small number of samples given an accuracy/probability, whereas our algorithm does exploration under a fixed budget of samples and guarantees *how good* the found action is. Even for simple multi-arm bandits, these two problems have different complexity (Carpentier & Locatelli, 2016) and can only be equivalent under unrealistic side knowledge (Gabillon et al., 2012). These related algorithms are also impractical for our setting. `TrailBlazer` uses confidence bounds that are humongous and `StOP` takes exponential time. Similar to `OLOP`, both also need to know noise and reward ranges.

2. Background

We model our problem with an MDP with state space X, action space A and dynamics such that taking the chosen action a_t at time t deterministically transitions the system from $x_t \in X$ to state $x_{t+1} = f(x_t, a_t)$ generating a reward $r_t = r(x_t, a_t) + \varepsilon_t$, with ε_t being the noise. We consider:

deterministic rewards The evaluations are noiseless, that is for all t, $\varepsilon_t \triangleq 0$ and $r_t = r(x_t, a_t)$.

stochastic rewards The evaluations are perturbed by a noise of range $b \in \mathbb{R}_+$: At any round, ε_t is a random variable, independent from noise at previous rounds,

$$\mathbb{E}\left[r_t | x_t\right] \triangleq r(x_t, a_t) \quad \text{and} \quad |r_t - r(x_t, a_t)| \leq b. \quad (1)$$

We assume that all rewards lie in the interval $[0, R_{\max}]$ and while the state space may be large and possibly infinite, that the action space is finite, with K available actions. We treat an infinite time-horizon problem with discounted rewards where the discount factor $(0 \leq \gamma < 1)$ is known. For any possible policy $\pi : X \to A$, we define the value function $V^\pi : X \to \mathbb{R}$ associated to π as $V^\pi(x) \triangleq \sum \gamma^t r(x_t, \pi(x_t))$, where x_t is the state of the system at time t when starting from x (i.e., $x_0 = x$) and following policy π. In the next definition, we also define the Q-value function $Q^\pi : X \times A \to \mathbb{R}$ associated to policy π, for each state-action pair (x, a), as the value of playing action a in state x and the following π thereafter.

Definition 1. *The Q-value function Q^π of policy π is*

$$Q^\pi(x, a) \triangleq r(x, a) + \gamma V^\pi(f(x, a)).$$

Notice that $V^{\pi}(x) = Q^{\pi}(x, \pi(x))$. We define the *optimal* value function, and Q-value function respectively, as $V^{\star}(x) \triangleq \sup_{\pi} V^{\pi}(x)$ and $Q^{\star}(x,a) \triangleq \sup_{\pi} Q^{\pi}(x,a)$, which corresponds to playing a first and optimally after. From the dynamic programming, we have the Bellman equations (Bertsekas & Tsitsiklis, 1996; Puterman, 1994),

$$V^{\star}(x) = \max_{a\in A} r(x,a) + \gamma V^{\star}(f(x,a)),$$
$$Q^{\star}(x,a) = r(x,a) + \gamma \max_{b\in A} Q^{\star}(f(x,a), b).$$

Let $[a:c] = \{a, a+1, \ldots, c\}$ with $a, c \in \mathbb{N}$, $a \le c$, and $[a] = [1:a]$ and let $\log_d$ be the logarithm in base d, $d \in \mathbb{R}$ and $\log$ without a subscript be the natural logarithm.

2.1. Optimistic planning under finite numerical budget

We assume that we have a generative model of f and r that generates simulated transitions and rewards. We want to make the best possible use of this model in order to recommend a best next action $a(n)$ such that the sum of the rewards resulting from playing $a(n)$ and then optimally afterwards is as close as possible to playing optimally from the beginning. For that purpose, we define the performance loss r_n as

$$r_n \triangleq \max_{a\in A} Q^{\star}(x,a) - Q^{\star}(x, a(n)).$$

2.2. The planning tree

For a given initial state x, consider the (infinite) planning tree defined by all possible sequences of actions (thus all possible reachable states starting from x). Let A^{∞} be the set of infinite sequences $(a_0, a_1, a_2, \ldots)$ where $a_t \in A$. The branching factor of this tree is the number of actions $|A| = K$. Since the dynamics are deterministic, to each finite sequence $a \in A^d$ of length d we assign a state that is reachable starting from x by following this sequence of d actions. Using standard notation for alphabets, we write $A^0 = \{\emptyset\}$ and $A^{\bullet}$ for the set of finite sequences. For $a \in A^{\bullet}$ we let $h(a)$ be the length of a, and $aA^h = \{aa', a' \in A^h\}$, where aa' denotes the sequence a followed by a'. We identify the set of finite sequences $a \in A^{\bullet}$ with the set of nodes of the tree. With $h' \le h$ and $a \in A^h$ we denote $a_{[h']}$ the sequence of action composed of the h' first actions from a, i.e., $\{a_0, \ldots, a_{h'-1}\}$. We fix $a_{[0]} \triangleq \emptyset$. The value $v(a)$ of an infinite sequence $a \in A^{\infty}$ is the discounted sum of rewards along the trajectory starting from the initial state x and defined by the choice of this sequence of actions,

$$v(a) \triangleq \sum_{t\ge 0} \gamma^t r(x_t, a_t), \text{where } x_0 = x; x_{t+1} = f(x_t, a_t)\cdot$$

Now, for any finite sequence $a \in A^{\bullet}$, or node, we define the value $v(a) = \sup_{a'\in A^{\infty}} v(aa')$. We write $v^{\star} = v(\emptyset) = \sup_{a\in A^{\infty}} v(a)$ for the optimal value at the initial state which is the root of the tree, $v^{\star} = V^{\star}(x)$. We denote the set of optimal infinite sequence of action as $A^{\star}$ which contains any $a \in A^{\infty}$ such that $v(a) = v^{\star}$. We note the set of optimal finite sequence of actions of depth h as $A^{\star,h}$ which contains any $a \in A^h$ such that $v(a) = v^{\star}$. We also define the u- and b-values for the lower- and upper- bounds on $v(a)$ as

$$u(a) \triangleq \sum_{t=0}^{h(a)-1} \gamma^t r(x_t, a_t), \text{and } b(a) = u(a) + \frac{\gamma^{h(a)} R_{\max}}{1-\gamma}\cdot$$

Indeed, since all rewards are in $[0, R_{\max}]$, we trivially have that $u(a) \le v(a) \le b(a)$. At any finite time t an algorithm has opened a set of nodes, which defines the expanded tree $\mathcal{T}_t$. We say the learner *opens* (or expands) a node a with m evaluations if uses the generative model f and r to generate m transitions and rewards for the K children nodes aA. In the deterministic reward feedback, $m = 1$. The bounds reported in this paper are in terms of the total number of openings n, instead of evaluations. The number of function evaluations is upper bounded by Kn. $T_{x,a}$ denotes the total number of evaluations allocated to action $a \in A$ in state x. We define, especially for the noisy case, the estimated value of the reward $\widehat{r}(x,a)$ of action $a \in A$ in state x. Given the $T_{x,a}$ evaluations $r_1, \ldots, r_{T_{x,a}}$, $\widehat{r}(x,a) \triangleq \frac{1}{T_{x,a}} \sum_{s=1}^{T_{x,a}} r_s$; the empirical average of rewards obtained at when performing action action $a \in A$ in state x. To ease notation, for $a \in A^m$ and $h \le m$, we write $T_a \triangleq \mathbb{E}[T_{x_{h(a)-1}, a_{h(a)-1}} | x_{t+1} \sim P(\cdot|x_t, a_t), x_0 = x]$ for the number of pulls to the last action in a. Similarly, $\widehat{r}_h(a) \triangleq \mathbb{E}[\widehat{r}(x_h, a_h) | x_{t+1} \sim P(\cdot|x_t, a_t), x_0 = x]$ and $r_h(a) \triangleq \mathbb{E}[r(x_h, a_h) | x_{t+1} \sim P(\cdot|x_t, a_t), x_0 = x]$. In the case of deterministic dynamics, x_h is such that $x_{t+1} \sim P(\cdot|x_t, a_t)$ and $x_0 = x$ is a fixed state from which we can sample from if we have a full access to the generative model. Hence, for a finite sequence $a \in A^{\bullet}$ or node,

$$\widehat{u}(a) \triangleq \sum_{t=0}^{h(a)-1} \gamma^t \widehat{r}(x_t, a_t) = \sum_{t=0}^{h(a)-1} \gamma^t \widehat{r}_t(a).$$

We the existence of at least one $a^{\star} \in A^{\infty}$ for which we have $V^{\star}(x) = \sup_{a\in A^{\infty}} v(a)$ and define a smoothness for v.

Proposition 1. *There exists* $\nu \in (0, R_{\max}/(1-\gamma)]$ *and* $\rho \in (0, \gamma]$ *such that* $\forall h \ge 0$, $\forall a \in A^h, u(a) \ge v(a) - \nu\rho^h$.

Note that this holds automatically for $\nu = R_{\max}/(1-\gamma)$ and $\rho = \gamma$. For some problems with an extra regularity this might also hold for some $\nu < R_{\max}/(1-\gamma)$ and $\rho < \gamma$. Note that our results automatically adapt to ρ without knowing its value. Note that while having a smoothness ρ means having rewards diminishing geometrically with depth with a ratio of ρ, the constant ν is linked to the scale of variation of the V and which can often be realistically smaller than $\nu < R_{\max}/(1-\gamma)$. We now define a measure of the quantity of near-optimal sequences for the smoothness ν, ρ.

Definition 2. *For any $\nu > 0$ and $\rho \in (0,1)$, the **branching factor** $\kappa^v(\nu,\rho)$ and with associated constant C, is*

$$\kappa^v(\nu,\rho) \triangleq \inf\{\kappa \geq 1 : \exists C > 1, \forall h \geq 0, \mathcal{N}_h^v(3\nu\rho^h) \leq C\kappa^h\},$$

where $\mathcal{N}_h^v(\varepsilon)$ is the number of nodes $a \in A^h$ of depth h such that $v(a) \geq v^\star - \varepsilon$.

We also define a related but different quantity than $\kappa^v(\nu,\rho)$. In particular, we define $\kappa^u(\nu,\rho)$ that uses $\mathcal{N}_h^u(\varepsilon)$ instead of $\mathcal{N}_h^v(\varepsilon)$ where $\mathcal{N}_h^u(\varepsilon)$ is the number of nodes $a \in A^h$ of depth h such that $u(a) \geq v^\star - \varepsilon$. Our results use the new quantity $\kappa^u(\nu,\rho)$, recover the previous results using $\kappa^v(\nu,\gamma)$ in the method of Bubeck & Munos (2010) and go beyond them. Indeed, theirs are only formulated for $\rho = \gamma$ while we prove the following claim in Appendix A.

Proposition 2. $\kappa^u(\nu/2,\gamma) \leq \kappa^v(\nu,\gamma) \leq \kappa^u(2\nu,\gamma)$.

3. Optimization vs planning

Our PlaTγPOOS approach is a very close sibling to the flat optimization algorithm, `StroquOOL` (Bartlett et al., 2019), however, we explain how the structure of the planning setting and the discount factor γ shaped our approach. The optimal application of an optimization algorithm to the planning setting is often not straightforward as discussed by Bubeck & Munos (2010). In their Section 2.2, Bubeck & Munos (2010) show that optimization can be applied to the planning problem either in a naïve way or a good way. The authors take as an example the uniform planning problem. The naïve and good strategies are evaluated by comparing the uncertainty $|u(a) - \widehat{u}(a)|$ of their estimates $\widehat{u}(a)$. Both strategies collect rewards identically, evaluating $u(a)$ for all the K^H nodes $a \in A^H$ at a fixed depth $H = h(a)$ by allocating one episode (of length H) for each a and receiving $r_{h,a}$, $1 \leq h \leq H$. In the naïve version, for all sequences a, the estimation of $u(a)$ uses only the H samples collected in the one episode related to a. Here $\widehat{u}(a) = \sum_{h=0}^{h(a)-1}\gamma^t r_{h,a}$ and $T_{a_{[h]}} = 1$. In contrast, the good planning strategy *reuses* estimates. For two distinct sequences a and a', the good strategy re-uses any samples $r_{h,a}$ for the estimation of both $u(a)$ and $u(a')$ if $a_{[h]} = a'_{[h]}$. Here $\widehat{u}(a) = \sum_{h=0}^{h(a)-1}\gamma^t \frac{1}{K^{H-h}}\sum_{a':a_{[h]}=a'_{[h]}} r_{h,a'}$ and $T_{a_{[h]}} = K^{H-h}$. This comparison is used to demonstrate the advantage of the use of the cross-sequence information to concentrate the estimate of the mean reward associated with each action more efficiently. It is by using this type of cross-sequence information that `OLOP` is able to obtain an improved regret over a naïve application of `HOO` (Bubeck et al., 2011) to the planning problem.

The previous discussion on cross-sequence information is tied to the case of uniform exploration strategies. Good uniform strategies guarantee $T_{a_{[h]}} = K^{H_u - h}$ by exploring until a reasonably shallow depth H_u but sampling all K^{H_u} nodes and sharing cross sequence information. In the case of `OLOP`, `HOO` or more particularly `StroquOOL`, only a subset S of size $|S| \ll K^{H_u}$ of the most promising nodes are explored but at deeper depth $H_s \gg H_u$. Therefore, obtaining for a given sub-sequence of actions a at depth $h < H_s$ a lower bound on the number of sequence of actions at depth H_s that contains a is complex in general. Actually one can design a problem with two actions that would drastically limit the amount of cross-sequence information for the optimal sequence of actions $a^\star$ in `StroquOOL`. Therefore applying `StroquOOL` with information sharing might not be enough and we chose to ensure algorithmicaly that a node at depth $h+1$ will be pulled γ^2 times less than a node at depth h. Indeed, using Chernoff-Hoeffding inequalities, we have $|u(a) - \widehat{u}(a)| \leq \sum_{h=0}^{h(a)-1}\gamma^h/\sqrt{T_{a_{[h]}}}$. However, PlaT$\gamma$POOS, through some simple reparametrization, remains very close to `StroquOOL` and we leave as an open question whether equivalent theoretical guarantees could be proved directly for `StroquOOL` applied to planning.

Input: n, A
Initialization: Open $t_{0,1}$. $h_{\max} = \lfloor n/\overline{\log}(n)\rfloor$.
For $h = 1$ to $h_{\max}$
  Open $\lfloor h_{\max}/h\rfloor$ nodes $a_{h,i}$ of depth h with largest values $u(a_{h,i})$.
Output $x(n) = \arg\max_{a_{h,i}:\in\mathcal{T}} u(a_{h,i})$.

Figure 1. Algorithm for free planning with no reset condition

4. Deterministic dynamics and rewards

In this section, we consider a simpler case of deterministic rewards in order to introduce our new ideas. The evaluations are noiseless, that is $\forall t$, $\varepsilon_t = 0$ and $r_t = r(x_t, a_t)$.

In Figure 1, we provide the `SequOOL` (Bartlett et al., 2019) algorithm applied to planning. In this case it is straightforward to follow the analysis of `SequOOL` in order to obtain the same rates, up to logarithmic factors, of simple regret as the state of the art algorithm `OPD` for the doubly deterministic case (Hren & Munos, 2008; Munos, 2014). and get the result[1] of Theorem 1. This direct usage was already discussed by Munos (2014, Section 5.1). Using `SequOOL` for planning already permits to have an algorithm that does not use the parameter $R_{\max}$ and that can adapt to extra smoothness in the value function ν, ρ as discussed Section 2. Note that to obtain similar adaptations to $R_{\max}$, ν and ρ , one could have already used `SOO` (Munos, 2014). However `SOO` *does not* come with optimal simple regret (Bartlett et al., 2019).

Theorem 1. *For any planning problem with associated (ν,ρ), and branching factor $\kappa = \kappa^u(\nu,\rho)$, we have, after n*

[1] $\overline{\log}(n)$ is the n-th harmonic number

Input: n, A

Initialization: Open $t_{0,1}$. $h_{\max} = \left\lfloor \frac{n}{\log n} \right\rfloor$.

For $h = 1$ to $h_{\max}$

Open $\lfloor h_{\max}/h^2 \rfloor$ nodes $a_{h,i}$ of depth h with largest values $u(a_{h,i})$.

Output $x(n) = \underset{a_{h,i} : \in \mathcal{T}}{\arg\max}\, u(a_{h,i})$.

Figure 2. Algorithm for constraint planning (restart)

rounds, the simple regret of SequOOL *bounded as:*

- *If* $\kappa = 1$, $r_n \leq \nu \rho^{\frac{1}{C}\left\lfloor \frac{n}{\log n} \right\rfloor}$.
- *If* $\kappa > 1$, $r_n = \mathcal{O}\left(\nu \left(\frac{\log \kappa}{C} \left\lfloor \frac{n}{\log n} \right\rfloor\right)^{-\frac{\log 1/\rho}{\log \kappa}}\right)$.

On the reset condition In practice, physical constraints can force the exploration to interact with the unknown environment under a reset condition. This means that there exists a starting state x and that a trajectory can only be sampled starting from this state and following a given policy. Therefore, if we wish to collect a sample from an arbitrary state x' after a reset, we must first reach that state from the starting state x.

SequOOL is a strategy that explores the MDP deeper and deeper from an initial starting state x. The original SequOOL strategy did not consider any reset condition, so to make a comparable analysis we will say that to 'open' a node a you first need to reach a at a budget cost of $h(a)$ which is equal to the depth of a. This additional cost has the consequence that SequOOL with reset will not be able to explore as deeply as without. A naïve extension is shown in Figure 2. Under a total limited budget of n, the number of nodes now open at depth h is $\mathcal{O}(n/h^2)$ instead of $\mathcal{O}(n/h)$ and the maximal depth is now of order of $\sqrt{n}$ instead of n. However, this does not influence the simple regret when $\kappa > 1$ by more than numerical constants as shown in Theorem 2. When $\kappa > 1$, the exponentially diminishing simple regret remains but is changed from ρ^n to $\rho^{\sqrt{n}}$.

Theorem 2. *For a planning problem with associated* (ν, ρ)*, and branching factor* $\kappa = \kappa^u(\nu, \rho)$*, after* n *rounds, the simple regret of* SequOOL *with reset condition verifies*

- *If* $\kappa = 1$, $r_n \leq \nu \rho^{\sqrt{\frac{1}{C}\left\lfloor \frac{n}{\log n} \right\rfloor}}$.
- *If* $\kappa > 1$, $r_n \leq \mathcal{O}\left(\nu \left(\frac{\log \kappa}{C} \left\lfloor \frac{n}{\log n} \right\rfloor\right)^{-\frac{\log 1/\rho}{\log \kappa}}\right)$.

5. Deterministic dynamics, stochastic rewards

We now describe the PlaTγPOOS algorithm detailed in Figure 3. In the presence of noise, it is natural to evaluate the cells multiple times, not just one time as in the deterministic case. The amount of times a cell should be evaluated to differentiate its value from the optimal value of the function depends on the gap between these two values as well as the range of noise. As we do not want to make *any* assumptions on knowing these quantities, our algorithm tries to be robust to any potential values by not making a fixed choice on the number of evaluations. Intuitively we do this following a path similar to StroquOOL (Bartlett et al., 2019) by using a modified version of SequOOL, denoted SequOOL(m), that allows us to evaluate cells m times, whereas for SequOOL, $m = 1$. Evaluating cells more times (m large) leads to a better quality of the mean estimates in each cell, however, as a trade-off, it uses more evaluations per depth. This would normally limit us from exploring deep depths of the partition, however, PlaTγPOOS takes advantage of the knowledge of γ which gives less weight to reward collected deeper in the tree. In order to obtain the concentration results for a node a on $\widehat{u}(a) - u(a)$ in Lemma 2, PlaTγPOOS uses a Chernoff-Hoeffding result that gives with high probability, $\widehat{u}(a) - u(a) \leq \sqrt{\sum_{h=0}^{h(a)-1} \gamma^{2h}/T_{a_{[h]}}}$ and balances the range of confidence intervals at different depths. Therefore, PlaTγPOOS tends to pull less with deeper depth as the number of pulls for a fixed m is $\lceil hm\gamma^{2h} \rceil$ where the additional h term ensures that the sum of confidence interval until depth h is bounded for any h. PlaTγPOOS then *implicitly* performs $\log n$ instances of SequOOL(m) each with a number of evaluations of $m = 2^p$, where we have $p \in [0 : \log n]$.

Input: n, A

Initialization: Open the root node $\emptyset$, $h_{\max}$ times.
$h_{\max} = \left\lfloor \frac{n}{2(\log_2 n+1)^2} \right\rfloor$, $p_{\max} = \lfloor \log_2 (h_{\max}) \rfloor$.

For $h = 1$ to $h_{\max}$ ◀ ***Exploration*** ▶

For $p = \lfloor \log_2(h_{\max}/\lceil h^2\gamma^{2h} \rceil) \rfloor$ down to 0

Open $\lceil h2^p\gamma^{2h} \rceil$ times the at most $\left\lfloor \frac{h_{\max}}{h\lceil h2^p\gamma^{2h} \rceil} \right\rfloor$ non-opened nodes $a^{h,i} \in A^h$ with highest values $\widehat{u}(a^{h,i})$ and given $T_{a^{h,i}} \geq \lceil (h-1)2^p\gamma^{2(h-1)} \rceil$.

For $p \in [0 : p_{\max}]$ ◀ ***Cross-validation*** ▶

Evaluate $(t+1)\gamma^{2t}h_{\max}(1-\gamma^2)^2$ times the actions at time t, a_t^p, of the *candidates:*

$a^p = \underset{a \in A^\bullet : \forall t \in [2:h(a)], T_{a_{[t]}} \geq \lceil (t-1)2^p\gamma^{2(t-1)} \rceil}{\arg\max} \widehat{u}(a)$.

Output $a^n = \underset{\{a^p, p \in [0:p_{\max}]\}}{\arg\max} \widehat{u}(a^p)$

Figure 3. The PlaTγPOOS algorithm

The 'Planning wiTh γ Plus an Online Optimization Strategy' algorithm PlaTγPOOS is in Figure 3. Remember that 'opening' a node means 'evaluating' its children actions. The algorithm opens nodes by sequentially diving them deeper and deeper from the root node $h = 0$ to a maximal depth of $h_{\max}$. At depth h, we allocate, in an al-most decreasing fashion, different number of evaluations $\lceil h2^p\gamma^{2h}\rceil$ to the nodes with highest value of that depth, with p starting at $\lfloor \log_2(h_{\max}/h)\rfloor$ down to 0. The best node that has been evaluated at least $\mathcal{O}(h_{\max}/h)$ times is opened with $\mathcal{O}(h_{\max}/h)$ evaluations, the two next best cells that have been evaluated at least $\mathcal{O}(h_{\max}/(2h)$ times are opened with $\mathcal{O}(h_{\max}/(2h))$ evaluations, the four next best cells that have been evaluated at least $\mathcal{O}(h_{\max}/(4h))$ times are opened with $\mathcal{O}(h_{\max}/(4h))$ evaluations and so on, until some $\mathcal{O}(h_{\max}/h)$ next best cells that have been evaluated at least once are opened with one evaluation. More precisely, given, p and h, we open, with $\lceil h2^p\gamma^{2h}\rceil$ evaluations, the $\lfloor h_{\max}/(h\lceil h2^p\gamma^{2h}\rceil)\rfloor$ non-previously-opened nodes $a^{h,i}\in A^h$ with highest values $\widehat{u}(a^{h,i})$ and given that $T_{a^{h,i}}\geq \lceil (h-1)2^p\gamma^{2(h-1)}\rceil$. The maximum number of evaluations of any node is $2^{p_{\max}}$, with $2^{p_{\max}}=\mathcal{O}(h_{\max})$ as $p_{\max}\triangleq \lfloor\log_2(h_{\max})\rfloor$. For each $p\in[0:p_{\max}]$, the candidate output a^p is the node a with highest estimated value such that all actions leading to that node have been evaluated in the following way $\forall t\in[2:h(a)], T_{a_{[t]}}\geq\lceil (t-1)2^p\gamma^{2(t-1)}\rceil$. We set $h_{\max}\triangleq\lfloor n/(2(\log_2 n+1)^2)\rfloor$.

5.1. Analysis of PlaTγPOOS

$\perp_{h,p}$ is the depth of the deepest opened node, a with at least $\lceil h2^p\gamma^h\rceil$ evaluations such that there is a $a^\star\in A^\star$ with $a^\star=ab$, with $b\in A^\infty$, at the end of the opening of depth h.

Lemma 1. *For any planning problem with associated (ν,ρ) (see Property 1), on event ξ defined in Appendix C, for any depths $h\in[h_{\max}]$, for any $p\in[0:\lfloor\log_2(h_{\max}/(h^2\gamma^{2h}))\rfloor]$, we have $\perp_{h,p}=h$ if conditions (1) and (2) simultaneously hold true.*
(1) $b\sqrt{\log(4n/\delta)/2^{p+1}}\leq\nu\rho^h$
(2) We distinguish cases and express the condition in each:

Case 1) $h2^p\gamma^{2h}\leq 1$*:*

$$\frac{h_{\max}}{h}=\frac{h_{\max}}{h\lceil h2^p\gamma^{2h}\rceil}\geq C\kappa(\nu,\rho)^h$$

And for all $h'\in[h]$, $\dfrac{h_{\max}}{h'^2 2^{p+1}\gamma^{2h'}}\geq C\kappa(\nu,\rho)^{h'}$.

Case 2) $h2^p\gamma^{2h}\geq 1$*:*

Case 2.1) $\gamma^2\kappa^u\geq 1$*:* $\dfrac{h_{\max}}{h^2 2^{p+1}\gamma^{2h}}\geq C\kappa(\nu,\rho)^h$

Case 2.2) $\gamma^2\kappa^u\leq 1$*:* $\dfrac{h_{\max}}{h^2 2^{p+1}}\geq C$.

Lemma 1 gives two conditions so that the cell containing a $a^\star\in A^\star$ is opened at depth h. This holds if (1) PlaTγPOOS opens, with $\lceil h2^p\gamma^{2h}\rceil$ evaluations, more cells at depth h than the number of near-optimal cells at depth h ($h_{\max}/\left(h^2 2^p\gamma^{2h}\right)\geq C\kappa(\nu,\rho)^h$ if $\gamma^2\kappa^u\geq 1$ and $h_{\max}/h2^p\geq C$ if $\gamma^2\kappa^u\leq 1$) and (2) the $\lceil h2^p\gamma^{2h}\rceil$ evaluations are sufficient to discriminate the empirical average of near-optimal cells from the empirical average of sub-optimal cells ($b\sqrt{\log(4n/\delta)/2^{p+1}}\leq\nu\rho^h$). To state the next theorems, we introduce $\widetilde{h}$, $\widetilde{h}_1$ and $\widetilde{h}_2$ three positive real numbers satisfying respectively the equations:

$$h_{\max}\nu^2\rho^{2\widetilde{h}_1}/\left(\widetilde{h}_1 b^2 g_{n,b}^{\delta,R_{\max}}\gamma^{2\widetilde{h}_1}\right)=C\kappa^{\widetilde{h}_1}\text{ and}$$

$$h_{\max}\nu^2\rho^{2\widetilde{h}_2}/\left(\widetilde{h}_2 b^2 g_{n,b}^{\delta,R_{\max}}\right)=C,\text{ where}$$

$$\widetilde{h}_1=\tfrac{1}{\log(\gamma^2\kappa/\rho^2)}\log\left(\tfrac{\overline{n}_1}{\log\overline{n}_1}\right)+o(1)\text{ and}$$

$$\widetilde{h}_2=\tfrac{1}{\log(1/\rho^2)}\log\left(\tfrac{\overline{n}_2}{\log\overline{n}_2}\right)+o(1)\text{ with}$$

$$\overline{n}_1\triangleq\tfrac{\nu^2 h_{\max}\log(\gamma^2\kappa/\rho^2)}{Cb^2 g_{n,b}^{\delta,R_{\max}}},\ \overline{n}_2\triangleq\tfrac{\nu^2 h_{\max}\log(1/\rho^2)}{Cb^2 g_{n,b}^{\delta,R_{\max}}},$$

where $g_{n,b}^{\delta,R_{\max}}\triangleq p_{\max}\log(R_{\max}n^{3/2}/b(1-\delta))$. $\widetilde{h}$ is defined similarly in Equation 5 in Appendix D. The quantities $\widetilde{h}_1$ and $\widetilde{h}_2$ give the respective depths of deepest cell opened by PlaTγPOOS that contains a $a^\star$ with high probability in the cases $\gamma^2\kappa\geq 1$ and $\gamma^2\kappa\leq 1$. Additionally, $\widetilde{h}_1$ and $\widetilde{h}_2$ also lets us characterize for which regime of the noise range b we recover results similar to the loss of the deterministic case. Discriminating on the noise regimes, we now state two of our results, Theorem 3 for a high noise and Theorem 4 for a low one. A more exhaustive list of results is in the Appendix D or in the Table 1.

Theorem 3. ***High-noise regime*** *After n rounds, for any problem with associated (ν,ρ), and branching factor denoted $\kappa=\kappa^u(\nu,\rho)$, if the noise b is high enough to verify both high-noise conditions as defined in the caption of Table 1, the simple regret of* PlaTγPOOS obeys

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

The proofs are in appendix D. They are quite technical but they are simply based on checking the conditions of Lemma 1 under different b,ρ,γ,κ regimes.

Theorem 4. ***Low-noise regime*** *After n rounds, for any problem with associated (ν,ρ), and branching factor denoted $\kappa=\kappa^u(\nu,\rho)$, if the noise b is low enough to verify both high-noise conditions as defined in the caption of Table 1, the simple regret of* PlaTγPOOS obeys

$$\mathbb{E}r_n=\begin{cases}\widetilde{\mathcal{O}}\left(\nu\rho^n\right) & \text{if }\kappa=1,\\ \widetilde{\mathcal{O}}\left(\nu\left(\frac{n}{b^2}\right)^{-\frac{\log(1/\rho)}{\log(\kappa)}}\right) & \text{if }\kappa>1.\end{cases}$$

	$\gamma^2\kappa \leq 1$		$\gamma^2\kappa \geq 1$	
	High noise (ii)	*Low noise* (ii)	*High noise* (iii)	*Low noise* (iii)
High noise (i)	$m_{\nu,\gamma}\left(\frac{n}{b^2}\right)^{-\frac{1}{2}}$	$\nu\rho^{\sqrt{n}}$	$m_{\nu,\gamma}\left(\frac{n}{b^2}\right)^{-\frac{\log(1/\rho)}{\log(\gamma^2\kappa/\rho^2)}}$	$\nu\left(\frac{n}{b^2}\right)^{-\frac{\log(1/\rho)}{\log(\kappa)}}$
Low noise (i)	$m_{\nu,\gamma}\left(\frac{n}{b^2}\right)^{-\frac{\log(1/\rho)}{\log(\kappa)}}$	$\kappa = 1 : \nu\rho^n$ $\kappa > 1 : \nu\left(\frac{n}{b^2}\right)^{-\frac{\log(1/\rho)}{\log(\kappa)}}$	$m_{\nu,\gamma}\left(\frac{n}{b^2}\right)^{-\frac{\log(1/\rho)}{\log(\kappa)}}$	$\nu\left(\frac{n}{b^2}\right)^{-\frac{\log(1/\rho)}{\log(\kappa)}}$

Table 1. Rates of the our upper bounds on the simple regret of PlaTγPOOS in various classes of problems. The condition on noise (i) is whether the noise b verifies $\nu^2\rho^{2\widetilde{h}}/\left(\gamma^{2\widetilde{h}}\widetilde{h}b^2 g_{n,b}^{\delta,R_{\max}}\right) \leq 1$ The condition on noise (ii) is whether $\nu^2\rho^{2\widetilde{h}_2}/\left(b^2 g_{n,b}^{\delta,R_{\max}}\right) \leq 1$. The condition on noise (iii) is whether $\nu^2\rho^{2\widetilde{h}_1}/\left(b^2 g_{n,b}^{\delta,R_{\max}}\right) \leq 1$. $m_{\nu,\gamma} = \max(1/1-\gamma^2, \nu)$.

Worst-case comparison with OLOP When b is large and known: In Table 1, we give our results in various classes of problems depending on the whether $\gamma^2\kappa \geq 1$, whether $\kappa = 1$ or $\kappa > 1$, and several conditions on the range of the noise b. The results for OLOP were distinguishing the results based on $\gamma^2\kappa$ being greater or smaller than 1. For this two cases we recover, as displayed in Table 1 with a grey background color, the same rate in term of n (for instance taking $b = 1$ like in OLOP and having $\rho = \gamma$) for the simple regret. However, we provide more specific treatment for sub-cases with associated improvements that we list and detail next.

Adaptation to the range of the noise b without a prior knowledge Our analysis shows that PlaTγPOOS adapts favorably to the unknown range of noise. Already, in the standard cases discussed above, where the noise is large, our bound already adapts to the amount of noise as it scales with n/b^2. OLOP requires an estimate $\widetilde{b}$ of b and has a regret scaling with $n/\widetilde{b}^2$ which is problematic in case of a wrong estimate $\widetilde{b} \gg b$. Moreover, we give technical conditions on the range of noise that shows when PlaTγPOOS gets improved rates. When $\gamma^2\kappa \geq 1$, OLOP was already obtaining rates that were the same as the rates of deterministic reward case. Therefore, beyond the n/b^2 improvement and the adaptation to extra smoothness that will be discussed latter, no more rate improvement should be expected. In the case $\gamma^2\kappa \leq 1$, the improvement are even more striking. When the noise is very low instead of the OLOP rate of $\mathcal{O}(1/\sqrt{n})$, we obtain the deterministic rate of OPD (Hren & Munos, 2008; Munos, 2014) which is either $n^{-\frac{\log(1/\rho)}{\log\kappa}}$ or ρ^n. These improved rates *could not* be obtained by OLOP. Indeed, OLOP relies on upper confidence bound (UCB) that uses a range of the noise $\widetilde{b}$ as input,

$$\overline{u}(a) = \mathcal{O}\left(\widehat{u}(a) + \sum_{h=0}^{h(a)-1}\gamma^h\widetilde{b}\sqrt{\frac{1}{T_{a_{[a]}}}} + \frac{R_{\max}\gamma^{h(a)}}{1-\gamma}\right).$$

This works if $\widetilde{b} = b$. However the true b is unknown. If $b = 0$, using any $\widetilde{b} > 0$ will not result in an improved rate.

Adaptation to additional smoothness ν and ρ beyond γ As defined in Section 2, we aim to adapt to the true smoothness ν, ρ of V which can go beyond γ. Our bounds show that PlaTγPOOS is able to take advantage of ν, ρ in a large portion of cases. In most cases in the rate the γ in OLOP is replaced by ρ in PlaTγPOOS. In the case where $\gamma^2\kappa \geq 1$ we have $r^{\texttt{OLOP}} = \mathcal{O}(n^{-\frac{\log(1/\gamma)}{\log(\kappa)}}) \leq \mathcal{O}(n^{-\frac{\log(1/\rho)}{\log(\gamma^2\kappa/\rho^2)}}) = r^{\texttt{PlaT}\gamma\texttt{POOS}}$.

Adaptation to the deterministic case and $\kappa = 1$ PlaTγPOOS adapts to the branching factor of the problem κ in a way that, under low noise conditions, leads to an exponentially decreasing simple regret $r_n^{\texttt{PlaT}\gamma\texttt{POOS}} = \mathcal{O}(\nu\rho^{\sqrt{n}})$. This is a light-years improvement over OLOP in these conditions as OLOP's regret is at best $r_n^{\texttt{PlaT}\gamma\texttt{POOS}} = \mathcal{O}(n^{-1/2})$. This result is possible because PlaTγPOOS can explore much deeper than OLOP, as its maximal depth is of order n. Actually, in most scenarios, the actual larger depth explored will be of order $\sqrt{n}$ due to sample limitations. On the other hand, OLOP can only go $\log n$ deep.

Moreover, $\kappa = 1$ is a common case in planning. Indeed, as discussed by Bubeck & Munos (2010), $\kappa = 1$ is equivalent to having near-optimal dimension $d = 0$ in an optimization task (Munos, 2014) which is a common value as shown by Valko et al. (2013). Therefore, we expect the case when $\gamma\kappa^2 \leq 1$, that is, where we get the most significant improvement other OLOP, to be the most common in practice.

The reset condition The effect of the reset condition, as discussed in Section 4, affects, in the stochastic reward case, PlaTγPOOS as follows. First, all polynomial rates stay the same. Then, only the exponential rates change. A ρ^n rate without the condition becomes a $\rho^{\sqrt{n}}$ with it. Next, a $\rho^{\sqrt{n}}$ rate becomes a $\rho^{n^{1/3}}$ one.

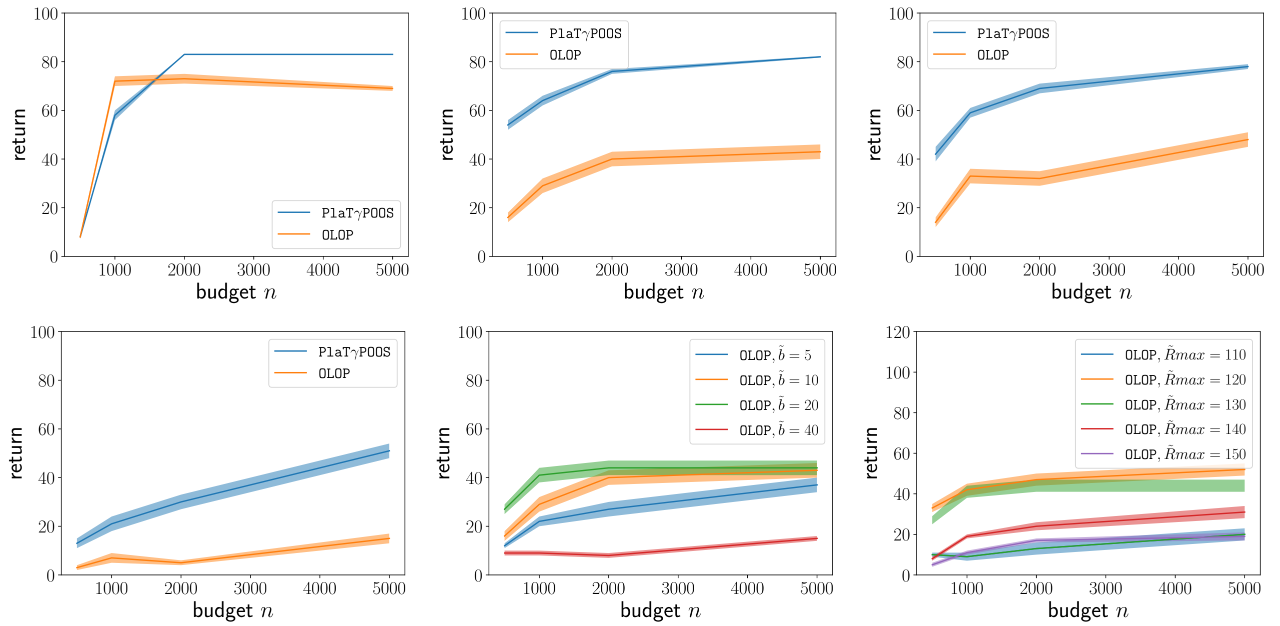

Figure 4. Top and bottom left: Expected cumulative discounted return collected by `OLOP` and `PlaTγPOOS` with different range of noise, $b = 1$ (top left), $b = 10$ (top center) $b = 20$, (top right) and $b = 50$ (bottom left). *Bottom center:* the sensitivity of `OLOP` to different $\widetilde{R}_{\max}$ parameters. *Bottom right:* the sensitivity of `OLOP` to different range of the input noise $\widetilde{b}$ as parameters while the true b is set to 10.

6. Numerical experiments

In this section, we empirically illustrate the benefits of `PlaTγPOOS`. We chose a simple MDP, shown in Figure 5. In this MDP, a state $x = (\text{bin}, d)$ is a pair of a binary variable bin and a non-negative integer d. The MDP has two actions that are also a binary variable a. If $\text{bin} \neq a$, the base reward is 2, in which case, the next state is $(a, 0)$. Otherwise, if $\text{bin} = a$, then $r = d$ and the next state is $(a, d+1)$. The reward is then shifted by adding 100 to it so that the noises with different ranges can be added on top without making the reward negative.

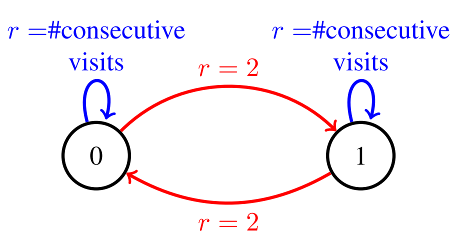

Figure 5. MDP used for experiments

The initial state is $(0, 0)$. Therefore, the agent has a choice. It can, for instance, remain in the same binary state *bin*, starting with a null reward but see its instant reward growing with time if it keeps taking the same action in the future. Alternatively, it could greedily switch to the other binary state bin and obtain a reward of 2 but delaying the hope of obtaining growing reward as in the first scenario. We set $\gamma = 0.95$. Therefore $R_{\max} \approx 130$.

Figure 4 reports the results. All the figures show the cumulative discounted return collected by `OLOP` and `PlaTγPOOS` after having interacted for 20 steps with the MDP, having chosen each time an action following their planning strategy and then being transferred to the state resulting of applying the recommended action in the current state and therefore also collecting a reward that is composing the final return.

Note that the return reported are shifted in order to not take into account the 100 fixed part of each the reward. The figures in the top row, as well as the figure at the bottom left, reports the comparison between the two returns of `OLOP` and `PlaTγPOOS` for different ranges of noise b. `PlaTγPOOS` is systematically outperforming `OLOP` while in this case `OLOP` is given as input parameters the correct $\widetilde{R}_{\max}$, with $\widetilde{R}_{\max} = R_{\max}$ and the correct range of the noise $\widetilde{b}$ as $\widetilde{b} = b$.

In Figure 4, bottom center and right, we illustrate the sensitivity of `OLOP` to misleading input parameters. Notice that the performance of `OLOP` is very vulnerable to these misspecifications while `PlaTγPOOS` is not using such inputs.

Acknowledgments The research presented was supported by European CHIST-ERA project DELTA, French Ministry of Higher Education and Research, Nord-Pas-de-Calais Regional Council, Inria and Otto-von-Guericke-Universität Magdeburg associated-team north-European project Allocate, and French National Research Agency project BoB (grant n.ANR-16-CE23-0003), FMJH Program PGMO with the support to this program from Criteo.

References

Bartlett, P. L., Gabillon, V., and Valko, M. A simple parameter-free and adaptive approach to optimization under a minimal local smoothness assumption. In *Algorithmic Learning Theory*, 2019.

Bertsekas, D. and Tsitsiklis, J. *Neuro-Dynamic Programming*. Athena Scientific, Belmont, MA, 1996.

Bubeck, S. and Munos, R. Open-loop optimistic planning. In *Conference on Learning Theory*, 2010.

Bubeck, S., Munos, R., Stoltz, G., and Szepesvári, C. X-armed bandits. *Journal of Machine Learning Research*, 12:1587–1627, 2011.

Buşoniu, L. and Munos, R. Optimistic planning for Markov decision processes. In *International Conference on Artificial Intelligence and Statistics*, 2012.

Carpentier, A. and Locatelli, A. Tight (lower) bounds for the fixed budget best-arm identification bandit problem. In *Conference on Learning Theory*, 2016.

Coquelin, P.-A. and Munos, R. Bandit algorithms for tree search. In *Uncertainty in Artificial Intelligence*, 2007.

Coulom, R. Efficient selectivity and backup operators in Monte-Carlo tree search. *Computers and games*, 4630: 72–83, 2007.

Feldman, Z. and Domshlak, C. Simple regret optimization in online planning for Markov decision processes. *Journal of Artificial Intelligence Research*, 2014.

Gabillon, V., Ghavamzadeh, M., and Lazaric, A. Best arm identification: A unified approach to fixed budget and fixed confidence. In *Neural Information Processing Systems*, 2012.

Gelly, S., Yizao, W., Munos, R., and Teytaud, O. Modification of UCT with patterns in Monte-Carlo Go. Technical report, Inria, 2006.

Grill, J.-B., Valko, M., and Munos, R. Blazing the trails before beating the path: Sample-efficient monte-carlo planning. In *Neural Information Processing Systems*, 2016.

Hoorfar, A. and Hassani, M. Inequalities on the lambert w function and hyperpower function. *Journal of Inequalities in Pure and Applied Mathematics*, 9(2):5–9, 2008.

Hren, J.-F. and Munos, R. Optimistic Planning of Deterministic Systems. In *European Workshop on Reinforcement Learning*, 2008.

Kaufmann, E. and Koolen, W. M. Monte-carlo tree search by best arm identification. In *Advances in Neural Information Processing Systems*, pp. 4897–4906, 2017.

Kocsis, L. and Szepesvári, C. Bandit-based Monte-Carlo planning. In *European Conference on Machine Learning*, 2006.

Leurent, E. and Maillard, O.-A. Practical open-loop optimistic planning. *arXiv preprint arXiv:1904.04700*, 2019.

Munos, R. Optimistic optimization of deterministic functions without the knowledge of its smoothness. In *Neural Information Processing Systems*, 2011.

Munos, R. From bandits to Monte-Carlo tree search: The optimistic principle applied to optimization and planning. *Foundations and Trends in Machine Learning*, 7(1):1–130, 2014.

Orabona, F. and Pál, D. Scale-free online learning. *Theoretical Computer Science*, 2018.

Orabona, F. and Tommasi, T. Backprop without Learning Rates Through Coin Betting. In *Neural Information Processing Systems*, 2017.

Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, New York, NY, 1994.

Ross, S., Mineiro, P., and Langford, J. Normalized online learning. In *Uncertainty in Artificial Intelligence*, may 2013.

Shah, D., Xie, Q., and Xu, Z. On reinforcement learning using monte carlo tree search with supervised learning: Non-asymptotic analysis. *arXiv preprint arXiv:1902.05213*, 2019.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

Szörényi, B., Kedenburg, G., and Munos, R. Optimistic planning in Markov decision processes using a generative model. In *Neural Information Processing Systems*, 2014.

Valko, M., Carpentier, A., and Munos, R. Stochastic simultaneous optimistic optimization. In *International Conference on Machine Learning*, 2013.

A. On the branching factor

Proposition 2. $\kappa^u(\nu/2, \gamma) \leq \kappa^v(\nu, \gamma) \leq \kappa^u(2\nu, \gamma)$.

Proof. We prove, for any global optimum, for $h \geq 0$, $\mathcal{N}_h^v(\varepsilon) \leq \mathcal{N}_h^u(\varepsilon + \gamma^h/1-\gamma)$. Let a node $a \in A^h$ such that $v(a) \geq v^\star - \varepsilon$ then we have $u(a) \geq v(a) - \gamma^h/1-\gamma \geq v^\star - \varepsilon - \gamma^h/1-\gamma$. Similarly we have, for any global optimum, for $h \geq 0$, $\mathcal{N}_h^u(\varepsilon) \leq \mathcal{N}_h^v(\varepsilon + \frac{\gamma^h}{1-\gamma})$. Using the Definition 2 we obtain the claimed result. □

B. PlaTγPOOS is not using a budget larger than $n+1$

Notice, for any given depth $h \in [1 : h_{\max}]$, PlaTγPOOS never uses more evaluations than $(p_{\max}+1)\frac{h_{\max}}{h}$ as

$$\begin{aligned}&\sum_{p=0}^{\lfloor \log_2(h_{\max}/\lceil h\gamma^{2h}\rceil)\rfloor} \left\lfloor \frac{h_{\max}}{h\lceil h2^p\gamma^{2h}\rceil} \right\rfloor \lceil h2^p\gamma^{2h}\rceil \\ &\leq \left(\left\lfloor \log_2(h_{\max}/\lceil h\gamma^{2h}\rceil)\right\rfloor + 1\right)\frac{h_{\max}}{h}\end{aligned}$$

Summing over the depths, PlaTγPOOS never uses more evaluations than the budget $n+1$ during its depth exploration as

$$\begin{aligned}&1 + (p_{\max}+1)\sum_{h=1}^{h_{\max}} \left\lfloor \frac{h_{\max}}{h} \right\rfloor \leq 1 + (p_{\max}+1)h_{\max}\sum_{h=1}^{h_{\max}} \frac{1}{h} \\ &= 1 + h_{\max}\overline{\log}(h_{\max})(p_{\max}+1) \leq 1 + h_{\max}(p_{\max}+1)^2 \\ &\leq n/2 + 1.\end{aligned}$$

We need to add the additional evaluation for the cross-validation at the end,

$$\sum_{p=0}^{p_{\max}}\sum_{t=0}^{h_{\max}} \frac{(t+1)\gamma^{2t}h_{\max}}{(1-\gamma^2)^2} \leq \sum_{p=0}^{p_{\max}} \left\lfloor \frac{n}{2(\log_2 n + 1)^2} \right\rfloor \leq \frac{n}{2}.$$

Therefore, in total the budget is not more than $n/2 + n/2 + 1 = n+1$. Again notice we use the budget of $n+1$ only for the notational convenience, we could also use $n/4$ for the evaluation in the end to fit under n (it's important that the amount of openings is *linear* in n).

C. Proofs of the lemmas

We first define and consider event ξ and prove it holds with high probability. The proof of the following lemmas can be found in Appendix C.

Lemma 2. *Let $\mathcal{C}$ be the set of sequence of actions evaluated by* PlaTγPOOS during one of its runs. $\mathcal{C}$ is a random quantity. Let ξ be the event under which all average estimates for the reward of the state-action pairs receiving at least one evaluation from PlaTγPOOS are within their classical confidence interval, then $P(\xi) \geq 1-\delta$, where

$$\begin{aligned}\xi \triangleq \Bigg\{&\forall a \in \mathcal{C}, \forall h \in [2 : h(a)],\, p \in [0 : p_{\max}] : \\ &\text{if } T_{a_{[h]}} \geq \left\lceil (h-1)2^p\gamma^{2(h-1)} \right\rceil, \\ &\text{then } |\widehat{u}(a) - u(a)| \leq b\sqrt{\frac{p_{\max}\log(4n/\delta)}{2^{p+1}}}\Bigg\}.\end{aligned}$$

Proof. The idea of the proof of this lemma follows the similar line as the proof of the statement given for StoSOO (Valko et al., 2013). The crucial point is that while we have potentially exponentially many combinations of cells that can be evaluated, given any particular execution we need to consider only a polynomial number of estimators, m, for which we can use a Azuma-Hoeffding concentration inequality.

We denote $\forall h \in [0:h_{\max}]$, $p \in [0:p_{\max}]: a^{i,h,p} \in \mathcal{C}$ the i-th evaluated node of depth h such that $\forall t \in [2:h], T_{a^{i,h,p}_{[t]}} \geq \lceil (t-1)2^p\gamma^{2(t-1)} \rceil$. Note that in PlaT$\gamma$POOS we have $T_{a^{i,h,p}_{[1]}} = h_{\max}$.

Though $a^{i,h,p}$ is random, we study the quantity $\left|\widehat{u}(a^{i,h,p}) - u(a^{i,h,p})\right|$. We recall that

$$\widehat{u}(a^{i,h,p}) - u(a^{i,h,p}) = \sum_{t=0}^{h-1} \gamma^t (\widehat{r}_t(a^{i,h,p}) - r_t(a^{i,h,p})) \tag{2}$$

$$= \sum_{t=0}^{h-1} \gamma^t \sum_{s=0}^{T_{a^{i,h,p}_{[t+1]}}} \frac{\widehat{r}^{i,h,p}_{t,s} - r_t(a^{i,h,p})}{T_{a^{i,h,p}_{[t+1]}}} \tag{3}$$

This quantity is composed of the elements $\widehat{r}^{i,h,p}_{t,s} - r_t(a^{i,h,p})$ that form a martingale.

Therefore using a Azuma-Hoeffding concentration inequality with a union bound already on the values of T we have

$$\mathbb{P}\left(\widehat{u}(a^{i,h,p}) - u(a^{i,h,p}) \leq b\sqrt{\sum_{t=0}^{h-1} \frac{\gamma^{2t}\log(p_{\max}/\delta)}{2T_{a^{i,h,p}_{[t+1]}}}}\right) \geq 1 - \delta/p_{\max}$$

Moreover we have for all $h \geq t > 1$,

$$\frac{\gamma^{2t}}{T_{a^{i,h,p}_{[t+1]}}} \leq \frac{\gamma^{2t}}{\lceil t2^p\gamma^{2t} \rceil} \leq \frac{\gamma^{2t}}{t2^p\gamma^{2t}} = \frac{1}{t2^p} \tag{4}$$

For $t = 0$, $\frac{\gamma^{2t}}{T_{a^{i,h,p}_{[t+1]}}} = \frac{1}{h_{\max}} \leq= \frac{1}{2^p}$ for all $p \leq p_{\max}$.

Therefore we have

$$\mathbb{P}\left(\widehat{u}(a^{i,h,p}) - u(a^{i,h,p}) \leq b\sqrt{\frac{\log h_{\max}\log(?/\delta)}{2^{p+1}}}\right) \geq 1 - \delta/?$$

Then we had an extra union bound other all cells that is bounded by n

□

Lemma 3. *For any planning problem with associated (ν, ρ) (see Property 1), on event ξ, for any depths $h \in [h_{\max}]$, for any $p \in [0: \lfloor \log_2(h_{\max}/(h^2\gamma^{2h})) \rfloor]$, we have $\perp_{h,p} = h$ if conditions (1) and (2) simultaneously hold true.*
(1) $b\sqrt{\log(4n/\delta)/2^{p+1}} \leq \nu\rho^h$
(2) For all $h' \in [h], h_{\max}/\left(h'\left\lceil h'2^p\gamma^{2h'}\right\rceil\right) \geq C\kappa(\nu,\rho)^{h'}$.
Finally we have $\perp_{0,p} = 0$.

Proof. We place ourselves on event ξ defined in Lemma 2 and for which we proved that $P(\xi) \geq 1 - \delta$. We fix p.

We prove the statement of the lemma, given that event ξ holds, by induction in the following sense. For a given h and p, we assume the hypotheses of the lemma for that h and p are true and we prove by induction that $\perp_{h',p} = h'$ for $h' \in [h]$.
1° For $h = 0$, we trivially have that $\perp_{h,p} \geq 0$.
2° Now consider $h' > 0$, and assume $\perp_{h'-1,p} = h' - 1$ with the objective to prove that $\perp_{h',p} = h'$.

Therefore, at the end of the processing of depth $h' - 1$, during which we were opening the nodes of depth $h' - 1$ we managed to open an optimal node that we denote $a^{\star,h'-1} \in A^{\star,h'-1}$. Moreover if we consider all the sequence of actions b that one can build by appending any action in A to $a^{\star,h'-1} \in A^{\star,h'-1}$, we have for all such b that $T_{b_{[t]}} \geq \lceil (t-1)2^p\gamma^{t-1} \rceil$ for $t \in [h']$.

Note that by definition there exist an optimal infinite sequence of actions $a^\star \in A^\star$ such that $a^{\star,h'-1} = a^\star_{[h'-1]}$

During phase h' the $\left\lfloor h_{\max}/\left(h'\left\lceil h'2^p\gamma^{2h'}\right\rceil\right)\right\rfloor$ evaluated nodes from $A^{h'-1}$ with highest values $\{\widehat{u}(a^{h'-1,i})\}_{h'-1,i}$ are opened.

For the purpose of contradiction, let us assume that $a^\star_{[h']}$ is not one of them. This would mean that there exist at least $\left\lfloor h_{\max}/\left(h'\left\lceil h'2^p\gamma^{2h'}\right\rceil\right)\right\rfloor$ nodes from $A^{h'}$, distinct from $a^\star_{[h']}$, satisfying $\widehat{u}(a^{h',i}) \geq \widehat{u}(a^\star_{[h']})$ and each verifying $T_{a^{h',i}_{[t]}} \geq \lceil t2^p\gamma^t\rceil$ for $t \in [h']$. This means that, for these nodes we have: $u(a_{h',i}) + \nu\rho^{h'} \geq u(a_{h',i}) + \nu\rho^h \overset{\textbf{(a)}}{\geq} u(a_{h',i}) + b\sqrt{\log(4n/\delta)/2^{p+1}} \overset{\textbf{(b)}}{\geq} \widehat{u}(a_{h',i}) \geq \widehat{u}(a_{h',i^\star_{h'}}) \overset{\textbf{(b)}}{\geq} u(a_{h',i^\star_{h'}}) - b\sqrt{\log(4n/\delta)/2^{p+1}} \overset{\textbf{(a)}}{\geq} u(a_{h',i^\star_{h'}}) - \nu\rho^h \geq u(a_{h',i^\star_{h'}}) - \nu\rho^{h'}$, where **(a)** is by assumption of the lemma, **(b)** is because ξ holds. As $u(a_{h',i^\star_{h'}}) \geq v^\star - \nu\rho^{h'}$ by Proposition 1, this means we have $\mathcal{N}^u_{h'}(3\nu\rho^{h'}) \geq \left\lfloor h_{\max}/\left(h'\left\lceil h'2^p\gamma^{h'}\right\rceil\right)\right\rfloor + 1$ (the $+1$ is for $a_{h,i^\star_h}$). However, by assumption of the lemma $h_{\max}/\left(h'\left\lceil h'2^p\gamma^{2h'}\right\rceil\right) \geq C\kappa(\nu,\rho)^{h'}$. It follows that in general $\mathcal{N}^u_{h'}(3\nu\rho^{h'}) > \left\lfloor C\kappa(\nu,\rho)^{h'}\right\rfloor$. This leads to having a contradiction with the $\kappa^u(\nu,\rho)$ with associated constant C as defined in Definition 2. Indeed, the condition $\mathcal{N}^u_{h'}(3\nu\rho^{h'}) \leq C\kappa(\nu,\rho)^{h'}$ in Definition 2 is equivalent to the condition $\mathcal{N}^u_{h'}(3\nu\rho^{h'}) \leq \left\lfloor C\kappa(\nu,\rho)^{h'}\right\rfloor$ as $\mathcal{N}^u_{h'}(3\nu\rho^{h'})$ is an integer.

□

Lemma 1. *For any planning problem with associated (ν,ρ) (see Property 1), on event ξ defined in Appendix C, for any depths $h \in [h_{\max}]$, for any $p \in [0 : \lfloor\log_2(h_{\max}/(h^2\gamma^{2h}))\rfloor]$, we have $\perp_{h,p} = h$ if conditions (1) and (2) simultaneously hold true.*
(1) $b\sqrt{\log(4n/\delta)/2^{p+1}} \leq \nu\rho^h$
(2) We distinguish cases and express the condition in each:

Case 1) $h2^p\gamma^{2h} \leq 1$***:***

$$\frac{h_{\max}}{h} = \frac{h_{\max}}{h\lceil h2^p\gamma^{2h}\rceil} \geq C\kappa(\nu,\rho)^h$$

And for all $h' \in [h]$, $\dfrac{h_{\max}}{h'^2 2^{p+1}\gamma^{2h'}} \geq C\kappa(\nu,\rho)^{h'}$.

Case 2) $h2^p\gamma^{2h} \geq 1$***:***

Case 2.1) $\gamma^2\kappa^u \geq 1$***:*** $\dfrac{h_{\max}}{h^2 2^{p+1}\gamma^{2h}} \geq C\kappa(\nu,\rho)^h$

Case 2.2) $\gamma^2\kappa^u \leq 1$***:*** $\dfrac{h_{\max}}{h^2 2^{p+1}} \geq C$.

Proof. To prove this statement we just need to show that we verify the hypotheses of Lemma 3. This means we need to prove that for all $h' \in [h]$, $h_{\max}/\left(h'\left\lceil h'2^p\gamma^{2h'}\right\rceil\right) \geq C\kappa(\nu,\rho)^{h'}$.

We first consider the case 2) where $h2^p\gamma^h \geq 1$. If $h = 1$ we already know $\perp_{0,p} \geq 0$. Let us now look at the case $h > 1$. First notice that $h2^p\gamma^{2h} \geq 1$ gives $(h-1)2^p\gamma^{2(h-1)} \geq 1$. If $\gamma^2\kappa^u \geq 1$ we have that for all $h' \in [h-1]$,

$$\frac{h_{\max}}{h'^2 2^{p+1}} \geq \frac{h_{\max}}{h^2 2^{p+1}} \geq C\left(\gamma^2\kappa(\nu,\rho)\right)^h \geq C\left(\gamma^2\kappa(\nu,\rho)\right)^{h'}$$

If $h > 1$, and if $\gamma^2\kappa^u \leq 1$ we have that for all $h' \in [h-1]$,

$$\frac{h_{\max}}{h'2^{p+1}} \geq \frac{h_{\max}}{h2^{p+1}} \geq C \geq C\left(\gamma^2\kappa(\nu,\rho)\right)^{h'}.$$

For both $\gamma^2\kappa^u \leq 1$ and $\gamma^2\kappa^u \geq 1$, we then have,

$$\frac{h_{\max}}{h'\lceil h'2^p\gamma^{2h'}\rceil} \geq \frac{h_{\max}}{h'h'2^{p+1}\gamma^{2h'}}$$

as $h2^p\gamma^{2h} \geq 1$.

For both $\gamma^2\kappa^u \leq 1$ and $\gamma^2\kappa^u \geq 1$, the previous equations mean that for $h' \in [h-1]$, h' verifies $h_{\max}/h'^2 2^{p+1}\gamma^{2h'} \geq C\kappa(\nu,\rho)^{h'} \geq 1$. Therefore $p \leq \left\lfloor \log_2(h_{\max}/(h'^2\gamma^{2h'}))\right\rfloor$.

We now consider case 1) where $h2^p\gamma^{2h} \leq 1$. We prove by induction that for all $h' \in [h], h_{\max}/\left(h'\left\lceil h'2^p\gamma^{2h'}\right\rceil\right) \geq C\kappa(\nu,\rho)^{h'}$.

1° By assumption of the lemma we say: $h_{\max}/\left(h\left\lceil h2^p\gamma^{2h}\right\rceil\right) \geq C\kappa(\nu,\rho)^h$

2° We further assume $h_{\max}/\left(h'\left\lceil h'2^p\gamma^{2h'}\right\rceil\right) \geq C\kappa(\nu,\rho)^{h'}$ is true for some $h' \leq h$ with $h'2^p\gamma^{2h'} \leq 1$

We want to prove that either:

both $(h'-1)2^p\gamma^{2(h'-1)} \leq 1$

and $\frac{h_{\max}}{(h'-1)\left\lceil (h'-1)2^p\gamma^{2(h'-1)}\right\rceil} \geq C\kappa(\nu,\rho)^{h'-1}$

or $(h'-1)2^p\gamma^{2(h'-1)} \geq 1$

then $h_{\max}/\left((h'')\left\lceil (h'')2^p\gamma^{2(h'')}\right\rceil\right) \geq C\kappa(\nu,\rho)^{h''}$ is already true for all $h'' \in [h']$. If $\left\lceil (h'-1)2^p\gamma^{2(h'-1)}\right\rceil = 1$ then we have

$$\frac{h_{\max}}{(h'-1)} \geq \frac{h_{\max}}{h'} \geq C\kappa(\nu,\rho)^{h'} \geq C\kappa(\nu,\rho)^{h'-1}$$

If $\left\lceil (h'-1)2^p\gamma^{2(h'-1)}\right\rceil > 1$, then we have that

$$h_{\max}/\left(h'\left\lceil (h'-1)2^p\gamma^{2(h'-1)}\right\rceil\right) \geq$$
$$h_{\max}/\left((h'-1)^2 2^{p+1}\gamma^{2(h'-1)}\right) \geq C\kappa(\nu,\rho)^{h'-1}$$

Using this inequality we can now use Case 2) to have that:
$h_{\max}/\left((h'')\left\lceil (h'')2^p\gamma^{2(h'')}\right\rceil\right) \geq C\kappa(\nu,\rho)^{h''}$ is already true for all $h'' \in [h']$.

The previous equations mean that for $h' \in [h-1]$, h' verifies $h_{\max}/\left(h'^2 2^{p+1}\gamma^{2h'}\right) \geq C\kappa(\nu,\rho)^{h'} \geq 1$. Therefore $p \leq \left\lfloor \log_2(h_{\max}/(h'^2\gamma^{2h'}))\right\rfloor$. □

D. Proof of Theorem 3 and Theorem 4

Theorem 3. *High-noise regime After n rounds, for any problem with associated (ν,ρ), and branching factor denoted $\kappa = \kappa^u(\nu,\rho)$, if the noise b is high enough to verify both high-noise conditions as defined in the caption of Table 1, the simple regret of* `PlaTγPOOS` obeys

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

Theorem 4. *Low-noise regime After n rounds, for any problem with associated (ν,ρ), and branching factor denoted $\kappa = \kappa^u(\nu,\rho)$, if the noise b is low enough to verify both high-noise conditions as defined in the caption of Table 1, the simple regret of* `PlaTγPOOS` obeys

$$\mathbb{E}r_n = \begin{cases} \widetilde{\mathcal{O}}\left(\nu\rho^n\right) & if\ \kappa = 1, \\ \widetilde{\mathcal{O}}\left(\nu\left(\frac{n}{b^2}\right)^{-\frac{\log(1/\rho)}{\log(\kappa)}}\right) & if\ \kappa > 1. \end{cases}$$

Proof of Theorem 3 and Theorem 4. We first place ourselves on the event ξ defined in Lemma 2 and where it is proven that $P(\xi) \geq 1-\delta$. We bound the simple regret of PlaTγPOOS on ξ.

Step 1) General definition of the regret

We chose $\delta = \frac{4b(1-\delta)}{R_{\max}\sqrt{n}}$ for the bound. We consider a problem with associated (ν, ρ). For simplicity we write $\kappa = \kappa^u(\nu, \rho)$. We have for all $p \in [0 : p_{\max}]$

$$
\begin{aligned}
& v(a^n) + \frac{b}{1-\gamma^2}\sqrt{\frac{p_{\max}\log(R_{\max}n^{3/2}/b)}{2h_{\max}}} \\
&\quad \geq u(a^n) + \frac{b}{1-\gamma^2}\sqrt{\frac{p_{\max}\log(R_{\max}n^{3/2}/b)}{2h_{\max}}} \overset{\textbf{(a)}}{\geq} \widehat{u}(a^n) \\
&\quad \overset{\textbf{(c)}}{\geq} \widehat{u}(a^p) \overset{\textbf{(b)}}{\geq} \widehat{u}\left(a^p_{[\perp_{h_{\max},p}+1]}\right) \\
&\quad \overset{\textbf{(a)}}{\geq} u\left(a^p_{[\perp_{h_{\max},p}+1]}\right) - \frac{b}{1-\gamma^2}\sqrt{\frac{p_{\max}\log(R_{\max}n^{3/2}/b)}{2h_{\max}}} \\
&\quad \overset{\textbf{(d)}}{\geq} v^\star - \nu\rho^{\perp_{h_{\max},p}+1} - \frac{b}{1-\gamma^2}\sqrt{\frac{p_{\max}\log(R_{\max}n^{3/2}/b)}{2h_{\max}}}
\end{aligned}
$$

where **(a)** is because the actions at time t, $a_t(n,p)$, of the *candidate* $a(n,p)$ have been evaluated $\frac{(t+1)\gamma^{2t}h_{\max}}{(1-\gamma)^2}$ times and because ξ holds, **(b)** is because $a^p_{[\perp_{h_{\max},p}+1]} \in \{a \in A^\bullet : \forall t \in [2 : h(a)], T_{a_{[t]}} \geq \lceil (t-1)2^p\gamma^{2(t-1)} \rceil\}$ and $a^p = \underset{a \in A^\bullet : \forall t \in [2:h(a)], T_{a_{[t]}} \geq \lceil (t-1)2^p\gamma^{2(t-1)}\rceil}{\arg\max} \widehat{u}(a)$, **(c)** is because $a^n = \underset{\{a^p, p \in [0:p_{\max}]\}}{\arg\max} \widehat{u}(a^p)$, and **(d)** is by Assumption 1.

From the previous inequality we have $r_n = v^\star - Q^\star(x, a^n) \leq \nu\rho^{\perp_{h_{\max},p}+1} + 2\frac{b}{1-\gamma^2}\sqrt{\frac{p_{\max}\log(R_{\max}n^{3/2}/b)}{2h_{\max}}}$, for $p \in [0 : p_{\max}]$.

Step 2) Defining some important depths For the rest of proof we want to lower bound $\max_{p \in [0:p_{\max}]} \perp_{h_{\max},p}$. Lemma 3 and 1 provide some sufficient conditions on p and h to get lower bounds. These conditions are inequalities in which as p gets smaller (fewer samples) or h gets larger (more depth) these conditions are more and more likely not to hold. For our bound on the regret of PlaTγPOOS to be small, we want quantities p and h where the inequalities hold but using as few samples as possible (small p) and having h as large as possible. Therefore we are interested in determining when the inequalities flip signs which is when they turn to equalities. This is what we solve next.

We set the notation $g^{\delta, R_{\max}}_{n,b} = p_{\max}\log(R_{\max}n^{3/2}/b(1-\delta))$.

In its most general form we are interested in the real numbers $\widetilde{h}$ and $\widetilde{p}$ are such that $\widetilde{h}$ is the larger real number such that for all $h \leq \widetilde{h}'$

$$
\frac{h_{\max}}{h^2 2^{\widetilde{p}+1}\gamma^{2h}} \geq C\kappa(\nu,\rho)^h \text{ while } \quad b\sqrt{\frac{g^{\delta,R_{\max}}_{n,b}}{2^{\widetilde{p}}}} = \nu\rho^{\widetilde{h}} \tag{5}
$$

In the case $\gamma^2\kappa \geq 1$ we can simply solve the following equations. We denote $\widetilde{h}_1, \widetilde{p}_1$ the real numbers satisfying

$$
\frac{h_{\max}\nu^2\rho^{2\widetilde{h}_1}}{2\widetilde{h}_1^2 b^2 g^{\delta,R_{\max}}_{n,b}\gamma^{2\widetilde{h}_1-2}} = C\kappa^{\widetilde{h}_1} \quad \text{and} \quad b\sqrt{\frac{g^{\delta,R_{\max}}_{n,b}}{2^{\widetilde{p}_1}}} = \nu\rho^{\widetilde{h}_1}\gamma^2. \tag{6}
$$

In the case $\gamma^2\kappa \leq 1$ the previous equation can possess two solutions where the largest of these two solutions will not verify Equation 5. Additionally the smallest solution might be hard to express in a close form when $\gamma^2\kappa \leq 1$. Therefore for simplicity we define for the case $\gamma^2\kappa \leq 1$, $\widetilde{h}_2, \widetilde{p}_2$ the real numbers satisfying

$$\frac{h_{\max}\nu^2\rho^{2\widetilde{h}_2}}{2\widetilde{h}_2^2 b^2 g_{n,b}^{\delta,R_{\max}}} = C \quad \text{and} \quad b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\widetilde{p}_2}}} = \nu\rho^{\widetilde{h}_2}. \tag{7}$$

$\widetilde{h}_1$ and $\widetilde{p}_1$ are defined for the case $\gamma^2\kappa \geq 1$ while $\widetilde{h}_2$ and $\widetilde{p}_2$ are defined for the case $\gamma^2\kappa \leq 1$. Our approach is to solve Equation 6 and 7 and then verify that it gives a valid indication of the behavior of our algorithm in term of its optimal p and h. We have

$$\widetilde{h}_1 = \frac{2}{\log(\gamma^2\kappa/\rho^2)} W\left(\log(\gamma^2\kappa/\rho^2)/2\sqrt{\frac{\gamma^2\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}\right)$$

$$\widetilde{h}_2 = \frac{2}{\log(1/\rho^2)} W\left(\log(1/\rho^2)/2\sqrt{\frac{\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}\right)$$

where standard W is the Lambert W function.

However after a close look at the Equation 7, we notice that it is possible to get values of $\widetilde{p}$ and $\widetilde{h}$ which would lead to a number of evaluations $\widetilde{h}2^{\widetilde{p}}\gamma^{\widetilde{h}} < 1$. This actually corresponds to an interesting case when the noise has a small range and where we can expect to obtain an improved result, that is: obtain a regret rate close to the deterministic case. This low range of noise case then has to be considered separately.

Therefore, we distinguish two cases which corresponds to different noise regimes depending on the value of b. Looking at the equation on the right of (7), we have that $\widetilde{h}2^{\widetilde{p}}\gamma^{\widetilde{h}} < 1$ if $\frac{\nu^2\rho^{2\widetilde{h}}}{\gamma^{2\widetilde{h}}\widetilde{h}b^2 g_{n,b}^{\delta,R_{\max}}} > 1$. Based on this condition we now consider the two cases. However for both of them we define some generic $\ddot{h}$ and $\ddot{p}$.

Case 1) $\gamma^2\kappa \geq 1$: Note that in this case then $\kappa > 1$. We subdivide this case into multiple subcases:

Case 1.1) Noise regime $\frac{\nu^2\rho^{2\widetilde{h}_1}}{\gamma^{2\widetilde{h}_1}\widetilde{h}_1 b^2 g_{n,b}^{\delta,R_{\max}}} \leq 1$

Case 1.1.1) High-noise regime $\frac{\nu^2\rho^{2\widetilde{h}_1}}{b^2 g_{n,b}^{\delta,R_{\max}}} \leq 1$

In this case, we denote $\ddot{h}_1 = \widetilde{h}_1$ and $\ddot{p}_1 = \widetilde{p}_1$. As $\frac{\nu^2\rho^{2\widetilde{h}_1}}{b^2 g_{n,b}^{\delta,R_{\max}}} \leq 1$ by construction, we have $\widetilde{p}_1 \geq 0$. Using standard properties of the $\lfloor\cdot\rfloor$ function, we have

$$b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\lfloor\widetilde{p}_1\rfloor+1}}} \leq b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\widetilde{p}_1}}} \leq \nu\rho^{\widetilde{h}_1} \leq \nu\rho^{\lfloor\widetilde{h}_1\rfloor} \tag{8}$$

$$\begin{aligned}
\text{and,}\quad & \frac{h_{\max}}{\left\lfloor\widetilde{h}_1\right\rfloor\left\lfloor\widetilde{h}_1\right\rfloor 2^{\lfloor\widetilde{p}_1\rfloor+1}\gamma^{2\lfloor\widetilde{h}_1\rfloor}} \geq \frac{h_{\max}}{\left\lfloor\widetilde{h}_1\right\rfloor\left\lfloor\widetilde{h}_1\right\rfloor 2^{\widetilde{p}_1+1}\gamma^{2\lfloor\widetilde{h}_1\rfloor}} \\
\geq\ & \frac{h_{\max}}{\left\lfloor\widetilde{h}_1\right\rfloor\widetilde{h}_1 2^{\widetilde{p}_1+1}\gamma^{2\widetilde{h}_1-2}} = \frac{h_{\max}\nu^2\rho^{2\widetilde{h}_1}}{2\widetilde{h}_1^2 b^2 g_{n,b}^{\delta,R_{\max}}\gamma^{2\widetilde{h}_1}} \\
=\ & C\kappa^{\widetilde{h}_1} \geq C\kappa^{\lfloor\widetilde{h}_1\rfloor}.
\end{aligned}$$

We will verify that $\left\lfloor\ddot{h}\right\rfloor$ is a reachable depth by PlaTγPOOS in the sense that $\ddot{h} \leq h_{\max}$ and $\lfloor\ddot{p}\rfloor \leq \left\lfloor\log_2(h_{\max}/(h^2\gamma^{2h}))\right\rfloor$ and . As $\kappa < 1$, and $\ddot{h} \geq 0$ we have $\kappa^{\ddot{h}} \geq 1$. This gives $C\kappa^{\ddot{h}} \geq 1$. Finally as $\frac{h_{\max}}{\ddot{h}^2 2^{\ddot{p}}\gamma^{2\ddot{h}}} \geq C\kappa^{\ddot{h}}$, we have $\ddot{h}^2\gamma^{2\ddot{h}} \leq h_{\max}/2^{\ddot{p}}$.

Case 1.1.2) Low-noise regime 1 $\frac{\nu^2\rho^{2\widetilde{h}_1}}{b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$

We denote $\ddot{h} = \overline{h}_1$ and $\ddot{p} = \overline{p}_1$ where $\overline{h}$ and $\overline{p}$ verify,

$$\frac{h_{\max}}{2\overline{h}_1^2\gamma^{2\overline{h}_1}} = C\kappa^{\overline{h}_1} \quad \text{and} \quad \overline{p}_1 = 0. \tag{9}$$

Again, $\frac{h_{\max}}{2\widetilde{h}_1^2 2^{p_1}\gamma^{2\widetilde{h}_1}} \geq 1$.

$$\overline{h}_1 = \frac{2}{\log(\gamma^2\kappa)} W\left(\log(\gamma^2\kappa)/2\sqrt{\frac{h_{\max}}{2C}}\right)$$

Using standard properties of the $\lfloor\cdot\rfloor$ function, we have

$$b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\lfloor\ddot{p}_1\rfloor+1}}} \leq b\sqrt{g_{n,b}^{\delta,R_{\max}}} < \nu\rho^{\widetilde{h}_1} \overset{\textbf{(a)}}{\leq} \nu\rho^{\overline{h}_1} \leq \nu\rho^{\lfloor\overline{h}_1\rfloor} \tag{10}$$

where **(a)** is because of the following reasoning. As we have $\frac{h_{\max}\nu^2\rho^{2\widetilde{h}_1}}{2\widetilde{h}_1^2 b^2 g_{n,b}^{\delta,R_{\max}}\gamma^{2\widetilde{h}_1}} = C\kappa^{\widetilde{h}_1}$ and $\frac{\nu^2\rho^{2\widetilde{h}_1}}{b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$, then, $\frac{h_{\max}}{2\widetilde{h}_1^2\gamma^{2\widetilde{h}_1}} \leq C\kappa^{\widetilde{h}_1}$. From the inequality $\frac{h_{\max}}{2\widetilde{h}_1^2} \leq C\kappa^{\widetilde{h}_1}\gamma^{2\widetilde{h}_1}$ and the fact that $\overline{h}_1$ corresponds to the case of equality $\frac{h_{\max}}{2\overline{h}_1^2} = C\kappa^{\overline{h}_1}\gamma^{2\overline{h}_1}$, we deduce that $\overline{h}_1 \leq \widetilde{h}_1$, since the left term of the inequality decreases with h while the right term increases (as $\gamma^2\kappa \geq 1$). Having $\overline{h}_1 \leq \widetilde{h}_1$ gives $\rho^{\overline{h}_1} \geq \rho^{\widetilde{h}_1}$.

Moreover, the term $\log(\gamma^2\kappa)$ of $\overline{h}_1$ could lead to think that we could potentially obtain a better rate that in the deterministic case where the term is $\log(\gamma^2\kappa)$. However this is not true because as $\overline{h}_1$ is the solution of $\overline{h}_1 = \frac{h_{\max}}{2\overline{h}_1^2\gamma^{2\overline{h}_1}} = C\kappa^{\overline{h}_1}$ and we have by assumption in this case $\overline{h}_1\gamma 2\overline{h}_1 \geq 1$ then $\overline{h}_1 \leq h_3$ where h_3 is defined as the solution of $h_3 = \frac{h_{\max}}{2\overline{h}_3\gamma^{2h_3}} = C\kappa^{h_3}$. We have $h_3 = \frac{1}{\log(\kappa)}W\left(\log(\kappa)\frac{h_{\max}}{2C}\right)$. Therefore one can see that this rate is not better that the deterministic rates.

Case 1.2) Low noise regime 2 $\frac{\nu^2\rho^{2\widetilde{h}_1}}{\gamma^{2\widetilde{h}_1}\widetilde{h}_1 b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$

We denote $\ddot{h} = \widehat{h}_1$ and $\ddot{p} = \widehat{p}_1$ where $\widehat{h}$ and $\widehat{p}$ verify,

$$\frac{h_{\max}}{2\widehat{h}_1} = C\kappa^{\widehat{h}_1} \quad \text{and} \quad \widehat{p}_1 = \max(0, \widetilde{p}_1)). \tag{11}$$

$$\widehat{h}_1 = \frac{1}{\log(\kappa)} W\left(\frac{h_{\max}\log(\kappa)}{2C}\right)$$

Using standard properties of the $\lfloor\cdot\rfloor$ function, we have

$$b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\lfloor\ddot{p}_1\rfloor+1}}} \leq b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\widehat{p}_1}}} < \nu\rho^{\widetilde{h}_1} \overset{\textbf{(a)}}{\leq} \nu\rho^{\widehat{h}_1} \leq \nu\rho^{\lfloor\widehat{h}_1\rfloor} \tag{12}$$

where **(a)** is because of the following reasoning. As we have $\frac{h_{\max}\nu^2\rho^{2\widetilde{h}_1}}{2\widetilde{h}_1^2 b^2 g_{n,b}^{\delta,R_{\max}}\gamma^{2\widetilde{h}_1}} = C\kappa^{\widetilde{h}_1}$ and $\frac{\nu^2\rho^{2\widetilde{h}_1}}{\gamma^{2\widetilde{h}_1}\widetilde{h}_1 b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$, then, $\frac{h_{\max}}{2\widetilde{h}_1} \leq C\kappa^{\widetilde{h}_1}$. From the inequality $\frac{h_{\max}}{2\widetilde{h}_1} \leq C\kappa^{\widetilde{h}_1}$ and the fact that $\widehat{h}_1$ corresponds to the case of equality $\frac{h_{\max}}{2\widehat{h}_1} = C\kappa^{\widehat{h}_1}$, we deduce that $\widehat{h}_1 \leq \widetilde{h}_1$, since the left term of the inequality decreases with h while the right term increases . Having $\widehat{h}_1 \leq \widetilde{h}_1$ gives $\rho^{\widehat{h}_1} \geq \rho^{\widetilde{h}_1}$.

Case 2) $\gamma^2\kappa \leq 1$

Case 2.1) Noise regime $\frac{\nu^2\rho^{2\widetilde{h}}}{\gamma^{2\widetilde{h}}\widetilde{h}b^2 g_{n,b}^{\delta,R_{\max}}} \leq 1$

Case 2.1.1) High-noise regime $\frac{\nu^2\rho^{2\widetilde{h}_2}}{b^2 g_{n,b}^{\delta,R_{\max}}} \leq 1$

In this case, we denote $\ddot{h} = \widetilde{h}_2$ and $\ddot{p} = \widetilde{p}_2$. As $\frac{\nu^2\rho^{2\widetilde{h}_2}}{b^2 g_{n,b}^{\delta,R_{\max}}} \leq 1$ by construction, we have $\widetilde{p}_2 \geq 0$. Using standard properties of the $\lfloor\cdot\rfloor$ function, we have

$$b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\lfloor\widetilde{p}_2\rfloor+1}}} \leq b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\widetilde{p}_2}}} = \nu\rho^{\widetilde{h}_2} \leq \nu\rho^{\lfloor\widetilde{h}_2\rfloor} \tag{13}$$

$$\text{and,}\quad \frac{h_{\max}}{\left\lfloor \widetilde{h}_2 \right\rfloor \left\lfloor \widetilde{h}_2 \right\rfloor 2^{\lfloor \widetilde{p}_2 \rfloor +1}} \geq \frac{h_{\max}}{\left\lfloor \widetilde{h}_2 \right\rfloor \left\lfloor \widetilde{h}_2 \right\rfloor 2^{\widetilde{p}_2 +1}}$$
$$\geq \frac{h_{\max}}{\widetilde{h}_2^2 2^{\widetilde{p}_2+1}} = \frac{h_{\max}\nu^2\rho^{2\widetilde{h}_2}}{2\widetilde{h}_2^2 b^2 g_{n,b}^{\delta,R_{\max}}}$$
$$= C.$$

We will verify that $\left\lfloor \ddot{h} \right\rfloor$ is a reachable depth by PlaTγPOOS in the sense that $\ddot{h} \leq h_{\max}$ and $\lfloor \ddot{p} \rfloor \leq \left\lfloor \log_2(h_{\max}/(h^2\gamma^{2h})) \right\rfloor$. As $\kappa < 1$, and $\ddot{h} \geq 0$ we have $\kappa^{\ddot{h}} \geq 1$. This gives $C\kappa^{\ddot{h}} \geq 1$. Finally as $\frac{h_{\max}}{\ddot{h}^2 2^{\ddot{p}}\gamma^{2\ddot{h}}} \geq C\kappa^{\ddot{h}}$, we have $\ddot{h}^2\gamma^{2\ddot{h}} \leq h_{\max}/2^{\ddot{p}}$.

Case 2.1.2) Low-noise regime 1 $\frac{\nu^2\rho^{2\widetilde{h}_2}}{b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$

We denote $\ddot{h} = \overline{h}_2$ and $\ddot{p} = \overline{p}_2$ where $\overline{h}$ and $\overline{p}$ verify,

$$\frac{h_{\max}}{2\overline{h}_2^2} = C \quad \text{and} \quad \overline{p}_2 = 0. \tag{14}$$

Again, $\frac{h_{\max}}{2\overline{h}_2^2 2^{p_2}\gamma^{2\overline{h}_2}} \geq 1$.

$$\overline{h}_2 = \sqrt{\frac{h_{\max}}{2C}}$$

Using standard properties of the $\lfloor\cdot\rfloor$ function, we have

$$b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\lfloor \ddot{p}_2 \rfloor+1}}} \leq b\sqrt{g_{n,b}^{\delta,R_{\max}}} < \nu\rho^{\widetilde{h}_1} \overset{\textbf{(a)}}{\leq} \nu\rho^{\overline{h}_2} \leq \nu\rho^{\lfloor \overline{h}_2 \rfloor} \tag{15}$$

where **(a)** is because of the following reasoning. As we have $\frac{h_{\max}\nu^2\rho^{2\widetilde{h}_2}}{2\widetilde{h}_2^2 b^2 g_{n,b}^{\delta,R_{\max}}} = C$ and $\frac{\nu^2\rho^{2\widetilde{h}_2}}{b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$, then, $\frac{h_{\max}}{2\widetilde{h}_2^2} \leq C$. From the inequality $\frac{h_{\max}}{2\widetilde{h}_2^2} \leq C$ and the fact that $\overline{h}_2$ corresponds to the case of equality $\frac{h_{\max}}{2\overline{h}_2^2} = C$, we deduce that $\overline{h}_2 \leq \widetilde{h}_2$, since the left term of the inequality decreases with h while the right term stays constant. Having $\overline{h}_2 \leq \widetilde{h}_2$ gives $\rho^{\overline{h}_2} \geq \rho^{\widetilde{h}_2}$.

Case 2.2) Low noise regime 2 $\frac{\nu^2\rho^{2\widetilde{h}}}{\gamma^{2\widetilde{h}}\widetilde{h}b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$

We denote $\ddot{h} = \widehat{h}_2$ and $\ddot{p} = \widehat{p}_2$ where $\widehat{h}$ and $\widehat{p}$ verify,

$$\frac{h_{\max}}{\widehat{h}_2} = C\kappa^{\widehat{h}_2} \tag{16}$$

$$\widehat{h}_2 = \frac{1}{\log(\kappa)} W\left(\frac{h_{\max}\log(\kappa)}{C}\right)$$

By construction, we have $\widetilde{h}_2 \leq \widetilde{h}$. We set

$$\widehat{p}_2 = \max(0, \widetilde{p})). \tag{17}$$

Using standard properties of the $\lfloor\cdot\rfloor$ function, we have

$$b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\lfloor \ddot{p}_2 \rfloor+1}}} \leq b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2^{\widetilde{p}}}} = \nu\rho^{\widetilde{h}} \overset{\textbf{(a)}}{\leq} \nu\rho^{\widehat{h}_2} \leq \nu\rho^{\lfloor \widehat{h}_2 \rfloor} \tag{18}$$

where **(a)** is because of the following reasoning. As we have $\frac{h_{\max}\nu^2\rho^{2\widetilde{h}}}{\widetilde{h}^2 b^2 g_{n,b}^{\delta,R_{\max}}\gamma^{2\widetilde{h}}} = C\kappa^{\widetilde{h}}$ and $\frac{\nu^2\rho^{2\widetilde{h}}}{\gamma^{2\widetilde{h}}\widetilde{h}b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$, then, $\frac{h_{\max}}{\widetilde{h}} \leq \kappa^{\widetilde{h}}$. From the inequality $\frac{h_{\max}}{\widetilde{h}} \leq C\kappa^{\widetilde{h}}$ and the fact that $\widehat{h}_2$ corresponds to the case of equality $\frac{h_{\max}}{2\widehat{h}_2} = C\kappa^{\widehat{h}_2}$, we deduce that $\widehat{h}_2 \leq \widetilde{h}$, since the left term of the inequality decreases with h while the right term increases. Having $\widehat{h}_2 \leq \widetilde{h}$ gives $\rho^{\widehat{h}_2} \geq \rho^{\widetilde{h}}$.

Step 3 Given these particular definitions of $\ddot{h}$ and $\ddot{p}$ in two distinct cases we now bound the regret.

We always have $\perp_{h_{\max},\lfloor\ddot{p}\rfloor} \geq 0$. If $\ddot{h} \geq 1$, as discussed above $\left\lfloor\ddot{h}\right\rfloor \in [h_{\max}]$, therefore $\perp_{h_{\max},\lfloor\ddot{p}\rfloor} \geq \perp_{\lfloor\ddot{h}\rfloor,\lfloor\ddot{p}\rfloor}$, as $\perp_{\cdot,\lfloor p\rfloor}$ is increasing for all $p \in [0, p_{\max}]$. Moreover on event ξ, and for the cases 1.1.1, 1.1.2, 2.1.1 and 2.1.2 described above, $\perp_{\lfloor\ddot{h}\rfloor,\lfloor\ddot{p}\rfloor} = \left\lfloor\ddot{h}\right\rfloor$ because of Lemma 1 (Case 2)) which assumptions on $\left\lfloor\ddot{h}\right\rfloor$ and $\lfloor\ddot{p}\rfloor$ are verified in each cases as detailed above and, in general, $\left\lfloor\ddot{h}\right\rfloor \in \left[\lfloor h_{\max}/2^{\ddot{p}}\rfloor\right]$ and $\lfloor\ddot{p}\rfloor \in [0 : p_{\max}]$. So, for the aforementioned cases, we have $\perp_{\lfloor h_{\max}/2^{\ddot{p}}\rfloor,\lfloor\ddot{p}\rfloor} \geq \left\lfloor\ddot{h}\right\rfloor$. Very similarly cases 1.2 and 2.2. lead to $\perp_{\lfloor h_{\max}/2^{\ddot{p}}\rfloor,\lfloor\ddot{p}\rfloor} \geq \left\lfloor\ddot{h}\right\rfloor$ by using Lemma 1 (Case 1)).

We bound the regret now discriminating on whether or not the event ξ holds. We have

$$
\begin{aligned}
r_n &\leq (1-\delta)\left(\nu\rho^{\perp_{h_{\max},\ddot{p}}+1} + 2\frac{b}{1-\gamma^2}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2h_{\max}}}\right) \\
&\quad + \delta \times \frac{R_{\max}}{1-\gamma} \leq \nu\rho^{\perp_{h_{\max},\ddot{p}}+1} + 2\frac{b}{1-\gamma^2}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{2h_{\max}}} + \frac{4b}{\sqrt{n}} \\
&\quad \leq \nu\rho^{\perp_{h_{\max},\ddot{p}}+1} + 6\frac{b}{1-\gamma^2}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}}.
\end{aligned}
$$

We can now bound the regret in the two regimes.

Case 1) $\gamma^2\kappa \geq 1$ **:** Note that in this case then $\kappa > 1$. We subdivide this case into multiple subcases:

Case 1.1) Noise regime $\frac{\nu^2\rho^{2\widetilde{h}_1}}{\gamma^{2\widetilde{h}_1}\widetilde{h}_1 b^2 g_{n,b}^{\delta,R_{\max}}} \leq 1$

Case 1.1.1) High-noise regime In general,we have

$$
r_n \;\leq\; \nu\rho^{\frac{2}{\log(\gamma^2\kappa/\rho^2)}W\left(\log(\gamma^2\kappa/\rho^2)/2\sqrt{\frac{\gamma^2\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}\right)} + 6\frac{b}{1-\gamma}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}}.
$$

Moreover, as proved by Hoorfar & Hassani (2008), the Lambert $W(x)$ function verifies for $x \geq e$, $W(x) \geq \log\left(\frac{x}{\log x}\right)$. Therefore, if $\log(\gamma^2\kappa/\rho^2)/2\sqrt{\frac{\gamma^2\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}} > e$ we have,

$$
\begin{aligned}
&r_n - 6\frac{b}{1-\gamma}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \\
&\leq \nu\rho^{\frac{2}{\log(\gamma^2\kappa/\rho^2)}\left(\log\left(\frac{\log(\gamma^2\kappa/\rho^2)/2\sqrt{\frac{\gamma^2\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}}{\log\left(\log(\gamma^2\kappa/\rho^2)/2\sqrt{\frac{\gamma^2\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}\right)}\right)\right)} \\
&= \nu e^{\frac{1}{\log(\gamma^2\kappa/\rho^2)}\left(\log\left(\frac{\log^2(\gamma^2\kappa/\rho^2)/2\frac{\gamma^2\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}{\log^2\left(\log(\gamma^2\kappa/\rho^2)/2\sqrt{\frac{\gamma^2\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}\right)}\right)\right)\log(\rho)} \\
&= \nu\left(\frac{\log^2(\gamma^2\kappa/\rho^2)/2\frac{\gamma^2\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}{\log^2\left(\log(\gamma^2\kappa/\rho^2)/2\sqrt{\frac{\gamma^2\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}\right)}\right)^{\frac{\log(\rho)}{\log(\gamma^2\kappa/\rho^2)}}.
\end{aligned}
$$

Case 1.1.2) Low-noise regime 1 $\frac{\nu^2\rho^{2\widetilde{h}_1}}{b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$

$$r_n \;\leq\; \nu\rho^{\frac{2}{\log(\gamma^2\kappa)}W\left(\log(\gamma^2\kappa)/2\sqrt{\frac{h_{\max}}{2C}}\right)} + 6\frac{b}{1-\gamma}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}}.$$

Moreover, as proved by Hoorfar & Hassani (2008), the Lambert $W(x)$ function verifies for $x \geq e$, $W(x) \geq \log\left(\frac{x}{\log x}\right)$. Therefore, if $\log(\gamma^2\kappa)/2\sqrt{\frac{h_{\max}}{2C}} > e$ we have,

$$\begin{aligned}
& r_n - 6\frac{b}{1-\gamma}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \\
& \leq \nu\rho^{\frac{2}{\log(\gamma^2\kappa)}\left(\log\left(\frac{\log(\gamma^2\kappa)/2\sqrt{\frac{h_{\max}}{2C}}}{\log\left(\log(\gamma^2\kappa)/2\sqrt{\frac{h_{\max}}{2C}}\right)}\right)\right)} \\
& = \nu e^{\frac{1}{\log(\gamma^2\kappa)}\left(\log\left(\frac{\log^2(\gamma^2\kappa)/2\frac{h_{\max}}{2C}}{\log^2\left(\log(\gamma^2\kappa)/2\sqrt{\frac{h_{\max}}{2C}}\right)}\right)\right)\log(\rho)} \\
& = \nu\left(\frac{\log^2(\gamma^2\kappa)/2\frac{h_{\max}}{2C}}{\log^2\left(\log(\gamma^2\kappa)/2\sqrt{\frac{h_{\max}}{2C}}\right)}\right)^{\frac{\log(\rho)}{\log(\gamma^2\kappa)}}.
\end{aligned}$$

We have $6b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \leq 6\frac{\nu\rho^{\widetilde{h}_1}}{\sqrt{g_{n,b}^{\delta,R_{\max}}}}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \leq 6\nu\rho^{\widetilde{h}_1} \leq 6\nu\rho^{\overline{h}_1}$.

Therefore $r_n \leq \nu\rho^{\perp_{h_{\max},\overline{p}}+1} + 6b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \leq 7\nu\rho^{\overline{h}_1}$.

Case 1.2) Low noise regime 2 $\frac{\nu^2\rho^{2\widetilde{h}_1}}{\gamma^{2\widetilde{h}_1}\widetilde{h}_1 b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$

$$\begin{aligned}
& r_n - 6\frac{b}{1-\gamma}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \\
& \leq \nu\left(\frac{\frac{h_{\max}\log(\kappa)}{2C}}{\log\left(\frac{h_{\max}\log(\kappa)}{2C}\right)}\right)^{\frac{\log(\rho)}{\log(\kappa)}}.
\end{aligned}$$

Moreover if $\frac{\nu^2\rho^{2\widetilde{h}_1}}{b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$, we have $6b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \leq 6\frac{\nu\rho^{\widetilde{h}_1}}{\sqrt{g_{n,b}^{\delta,R_{\max}}}}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \leq 6\nu\rho^{\widetilde{h}_1} \leq 6\nu\rho^{\widehat{h}_1}$.

Therefore $r_n \leq \nu\rho^{\perp_{h_{\max},\overline{p}}+1} + 6b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \leq 7\nu\rho^{\widehat{h}_1}$.

Case 2) $\gamma^2\kappa \leq 1$

Case 2.1) Noise regime $\frac{\nu^2\rho^{2\widetilde{h}}}{\gamma^{2\widetilde{h}}\widetilde{h}b^2 g_{n,b}^{\delta,R_{\max}}} \leq 1$

Case 2.1.1) High-noise regime $\frac{\nu^2\rho^{2\widetilde{h}_2}}{b^2 g_{n,b}^{\delta,R_{\max}}} \leq 1$

$$
\begin{aligned}
&r_n - 6\frac{b}{1-\gamma}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}}\\
&\leq \nu\left(\frac{\log^2(1/\rho^2)/2\frac{\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}{\log^2\left(\log(1/\rho^2)/2\sqrt{\frac{\nu^2 h_{\max}}{2Cb^2 g_{n,b}^{\delta,R_{\max}}}}\right)}\right)^{-\frac{1}{2}}.
\end{aligned}
$$

Case 2.1.2) Low-noise regime 1 $\frac{\nu^2\rho^{2\widetilde{h}_2}}{b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$

Here with a similar reasoning as in the case 1.1.2) we have $r_n \leq 7\nu\rho^{\overline{h}_1} \leq 7\nu\rho^{\sqrt{\frac{h_{\max}}{2C}}}$.

Case 2.2) Low noise regime 2 $\frac{\nu^2\rho^{2\widetilde{h}}}{\gamma^{2\widetilde{h}}\widetilde{h}b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$

$$
\begin{aligned}
&r_n - 6\frac{b}{1-\gamma}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}}\\
&\leq \nu\left(\frac{\frac{h_{\max}\log(\kappa)}{C}}{\log\left(\frac{h_{\max}\log(\kappa)}{C}\right)}\right)^{\frac{\log(\rho)}{\log(\kappa)}}.
\end{aligned}
$$

Moreover if $\frac{\nu^2\rho^{2\widetilde{h}}}{b^2 g_{n,b}^{\delta,R_{\max}}} \geq 1$, we have $6b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \leq 6\frac{\nu\rho^{\widetilde{h}}}{\sqrt{g_{n,b}^{\delta,R_{\max}}}}\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \leq 6\nu\rho^{\widetilde{h}} \leq 6\nu\rho^{\widehat{h}_2}$.

Therefore $r_n \leq \nu\rho^{\perp_{h_{\max},\overline{p}}+1} + 6b\sqrt{\frac{g_{n,b}^{\delta,R_{\max}}}{h_{\max}}} \leq 7\nu\rho^{\widehat{h}_1}$.

Moreover if $\kappa = 1$ then $r_n \leq 7\nu\rho^{\frac{h_{\max}}{C}}$ □

E. Use of the budget

Remark 1. *The algorithm can be made anytime and unaware of n using the classic 'doubling trick'.*

Remark 2 (More efficient use of the budget)**.** *Because of the use of the floor functions $\lfloor\cdot\rfloor$, the budget used in practice can be significantly smaller than n. While this only affects numerical constants in the bounds, in practice, it can noticeably influence the performance. Therefore one should consider, for instance, having $h_{\max}$ replaced by $c \times h_{\max}$ with c been the largest number such that the budget is still smaller than n. Additionally, the use of the budget n could be slightly optimized by taking into account that the necessary number of pulls at depth h cannot be larger than K^h.*