Online Stochastic Optimization under Correlated Bandit Feedback

Mohammad Gheshlaghi Azar MOHAMMAD.AZAR@NORTHWESTERN.EDU
Rehabilitation Institute of Chicago, Northwestern University

Alessandro Lazaric ALESSANDRO.LAZARIC@INRIA.FR
Team SequeL, INRIA Nord Europe

Emma Brunskill EBRUN@CS.CMU.EDU
School of Computer Science, CMU

Abstract

In this paper we consider the problem of online stochastic optimization of a locally smooth function under bandit feedback. We introduce the high-confidence tree (HCT) algorithm, a novel anytime $\mathcal{X}$-armed bandit algorithm, and derive regret bounds matching the performance of state-of-the-art algorithms in terms of the dependency on number of steps and the near-optimality dimension. The main advantage of HCT is that it handles the challenging case of correlated bandit feedback (reward), whereas existing methods require rewards to be conditionally independent. HCT also improves on the state-of-the-art in terms of the memory requirement, as well as requiring a weaker smoothness assumption on the mean-reward function in comparison with the existing anytime algorithms. Finally, we discuss how HCT can be applied to the problem of policy search in reinforcement learning and we report preliminary empirical results.

1. Introduction

We consider the problem of maximizing the sum of the rewards obtained by sequentially evaluating an unknown stochastic function. This problem is known as stochastic optimization under bandit feedback or $\mathcal{X}$*-armed bandit*, since each function evaluation can be viewed as pulling one of the arms in a generic arm space $\mathcal{X}$. Our objective is to minimize the cumulative regret relative to selecting at each step the global maximum of the function. In particular, we focus on the case where the reward obtained by pulling an arm (i.e., evaluating the function in a point) may depend on prior history of evaluations and outcomes. This implies that the reward, conditioned on its corresponding arm, is not an independent and identically distributed (iid) random variable, in contrast to prior work on $\mathcal{X}$-armed bandits (see e.g., Munos, 2013; Kleinberg et al., 2013; Bubeck et al., 2011a). $\mathcal{X}$-armed bandit with correlated reward is relevant to many real-world applications, including internet auctions, adaptive routing, and online games. As one important example, we show that the problem of policy search in an ergodic Markov Decision Process (MDP), a popular setting for learning in unknown MDPs, can be framed as an instance of the setting we consider in this paper (Sect. 5).

Our approach builds on recent advances in $\mathcal{X}$-armed bandits for iid settings (Bull, 2013; Djolonga et al., 2013; Bubeck et al., 2011a; Srinivas et al., 2009; Cope, 2009; Kleinberg et al., 2008; Auer et al., 2007). Under regularity assumptions on the mean-reward function (e.g., Lipschitz-smoothness), these methods provide formal guarantees on the cumulative regret, which is proved to scale sub-linearly w.r.t. the number of steps n. To obtain this regret, these methods heavily rely on the iid assumption. To handle non-iid settings, we introduce a new anytime $\mathcal{X}$-armed bandit algorithm, called *high-confidence tree* (HCT) (Sect. 3). Similar to the HOO algorithm of Bubeck et al. (2011a), *HCT* makes use of a covering binary tree to explore the arm space. The tree is constructed incrementally in an optimistic fashion, where the exploration of the arm space is guided by upper bounds on the largest reward of the arms covered by a particular node. Our key insight is that to achieve small regret it is enough to expand an optimistic node only when the estimate of its mean-reward has become sufficiently accurate. Under mild ergodicity and mixing assumptions, this allows us to obtain an accurate estimate of the reward of a particular arm even in the non-iid setting. Despite handling a more general case of non-iid feedback, our regret bounds matches (Sect. 4.1) that of HOO (Bubeck et al., 2011a) and zooming algorithm

Proceedings of the 31^{st} International Conference on Machine Learning, Beijing, China, 2014. JMLR: W&CP volume 32.

(Kleinberg et al., 2008), both of which only apply to iid setting, in terms of dependency on the number of steps n and the near-optimality dimension d (Sect. 2). An important part of the proof of this result is the development of concentration inequalities for non-iid episodic random variables (Sect. 4). In addition to this result, the structure of our HCT approach has a favorable sub-linear space complexity of $O(n^{d/(d+2)}(\log n)^{2/(d+2)})$ and a linearithmic runtime complexity, making it suitable for scaling to *big data* scenarios. These results meet or improve the space and time complexity of prior work designed for iid data (Sect. 4.2). Finally, we demonstrate the benefit in simulations (Sect. 6).

2. Preliminaries

The optimization problem. Let X be a measurable space of arms. We formalize the optimization problem as an interaction between the learner and the environment. At each time step t, the learner pulls an arm x_t in X and the environment returns a reward $r_t \in [0,1]$ and possibly a context $y_t \in \mathsf{Y}$, with Y a measurable space (e.g., the state space of a Markov decision process). Whenever needed, we explicitly relate r_t to the arm pulled by using the notation $r_t(x)$. The context y_t and the reward r_t may depend on the history of all previous rewards, pulls, and contexts as well as the current pull x_t. For any time step $t > 0$, the space of histories $\mathsf{H}_t := ([0,1] \times \mathsf{X} \times \mathsf{Y})^t$ is defined as the space of past rewards, arms, and observations (with $\mathsf{H}_0 = \emptyset$). An environment M corresponds to an infinite sequence of time-dependent probability measures $M = (Q_1, Q_2, \dots)$, such that each $Q_t : \mathsf{H}_{t-1} \times \mathsf{X} \to \mathsf{M}([0,1] \times \mathsf{Y})$ is a mapping from the history H_{t-1} and the arm space X to the space of probability measures on rewards and contexts. Let $\mathsf{Z} = ([0,1] \times \mathsf{X} \times \mathsf{Y})$, at each step t we define the random variable $z_t = (r_t, x_t, y_t) \in \mathsf{Z}$ and we introduce the filtration F_t as a σ-algebra generated by $(z_1, z_2, \dots, z_t)$. At each step t, the arm x_t selected is F_{t-1}-measurable since it is based on all the information available up to time $t-1$. In general, the pulling strategy of the learner can be expressed as an infinite sequence of measurable mappings $(\psi_1, \psi_2, \dots)$, where $\psi_t : \mathsf{H}_{t-1} \to \mathsf{M}(\mathsf{X})$ maps H_{t-1} to the space of probability measures on arms. We now refine this general setting with two assumptions on the reward-generating process.

Definition 1 (Time average reward)**.** *For any $x \in \mathsf{X}$, $S > 0$ and $0 < s \le S$, the time average reward is $\bar{r}_{s\to S}(x) := 1/(S-s+1)\sum_{s'=s}^{S} r_{s'}(x)$.*

We now state our first assumption which guarantees that the mean of the process is well defined (ergodicity).

Assumption 1 (Ergodicity)**.** *For any $x \in \mathsf{X}$, any $s > 0$ and any sequence of prior pulls $(x_1, x_2, \dots, x_{s-1})$, the process $(z_t)_{t>0}$ is such that the mean-reward function $f(x) := \lim_{S\to\infty} \mathbb{E}(\bar{r}_{s\to S}(x) | \mathsf{F}_{s-1})$ exists.*

This assumption implies that, regardless of the history of observations F_{s-1}, if arm x is pulled infinitely many times from time s, then the time average reward converges in expectation to a fixed point which only depends on arm x and is independent from the past history F_{s-1}. We also make the following mixing assumption (see e.g., Levin et al., 2006, Chap. 4).

Assumption 2 (Finite mixing time)**.** *There exists a constant $\Gamma \ge 0$ (mixing time) such that for any $x \in \mathsf{X}$, any $S > 0$, any $0 < s \le S$ and any sequence of prior pulls $(x_1, x_2, \dots, x_{s-1})$, the process $(z_t)_{t>0}$ is such that we have that $|\mathbb{E}[\sum_{s'=s}^{S}(r_{s'}(x) - f(x))\big|\mathsf{F}_{s-1}]| \le \Gamma$.*

This assumption implies that the stochastic reward process induced by pulling arm x can not substantially deviate from $f(x)$ in expectation for more than Γ *transient* steps. Note that both assumptions trivially hold if each arm is an independent iid process: in this case $f(x)$ is the mean-reward of arm x and $\Gamma = 0$.

Given the mean-reward f, we assume that the maximizer $x^* = \arg\max_x f(x)$ exists and we denote the corresponding maximum $f(x^*)$ by f^*. We measure the performance of the learner over n steps by its regret R_n w.r.t. the f^*, defined as $R_n := nf^* - \sum_{t=1}^{n} r_t$. The goal of learner, at every $0 \le t \le n$, is to choose a strategy ψ_t such that the regret R_n is as small as possible.

Related models. Although the learner observes a context y_t at each time t, this problem differs from the contextual bandit setting (see e.g., Slivkins, 2009). In contextual bandits, the reward r_t is random realization of a function $r(x_t, y_t)$ of the selected arm and input context y_t. The contextual bandit objective is typically to minimize the regret against the optimal arm in the context provided at each step, y_t, i.e. $x_t^* = \arg\max r(x, y_t)$. A key difference is that in our model the reward, and next context, may depend on the entire history of rewards, arms pulled, and contexts, instead of only the current context and arm, and we define $f(x)$ only as the average reward obtained by pulling arm x. In this sense, our model is related to the reinforcement learning (RL) problem of trying to find a policy that maximizes the long run reward. Among prior work in RL our setting is similar to the general reinforcement learning model of Lattimore et al. (2013) which also considers arbitrary temporal dependence between rewards and observations. The main difference is that here we consider the regret in undiscounted reward scenario, whereas the focus of Lattimore et al. (2013) is on proving PAC-bounds in the discounted reward case. Another difference is that in our model, unlike that of Lattimore et al. (2013), the observation and action spaces need not to be finite (see further discussion in Sect. 5).

The cover tree. Similar to recent optimization methods (e.g., Bubeck et al., 2011a), our approach seeks to min-

imize the regret by building an estimate of f using an infinite binary *covering tree* T, in which each node covers a subset of X. We denote by (h, i) the node at depth h and index i among the nodes at the same depth (e.g., the root node which covers X is indexed by $(0, 1)$). By convention $(h+1, 2i-1)$ and $(h+1, 2i)$ refer to the two children of the node (h, i). The area corresponding to each node (h, i) is denoted by $\mathsf{P}_{h,i} \subseteq \mathsf{X}$. These regions must be measurable and, at each depth, they partition X with no overlap, i.e.,

$$\mathsf{P}_{0,1} = \mathsf{X}$$
$$\mathsf{P}_{h,i} = \mathsf{P}_{h+1,2i-1} \cup \mathsf{P}_{h,2i} \quad \forall h \geq 0 \text{ and } 1 \leq i \leq 2^h.$$

For each node (h, i), we define an arm $x_{h,i} \in \mathsf{P}_{h,i}$, which is pulled whenever the node (h, i) is selected.

We now state a few additional geometrical assumptions.

Assumption 3 (Dissimilarity)**.** *The space X is equipped with a dissimilarity function $\ell : \mathsf{X}^2 \to \mathbb{R}$ such that $\ell(x, x') \geq 0$ for all $(x, x') \in \mathsf{X}^2$ and $\ell(x, x) = 0$.*

Given a dissimilarity ℓ, the diameter of a subset $A \subseteq \mathsf{X}$ is defined as $\mathrm{diam}(A) := \sup_{x,y \in A} \ell(x, y)$, while an ℓ-ball of radius $\varepsilon > 0$ and center $x \in \mathsf{X}$ is defined as $\mathsf{B}(x, \varepsilon) := \{x' \in \mathsf{X} : \ell(x, x') \leq \varepsilon\}$.

Assumption 4 (Local smoothness)**.** *We assume that there exist constants $\nu_2, \nu_1 > 0$ and $0 < \rho < 1$ such that for all nodes (h, i):*

(a) $\mathrm{diam}(\mathsf{P}_{h,i}) \leq \nu_1 \rho^h$

(b) $\exists x^o_{h,i} \in \mathsf{P}_{h,i}$ s.t. $\mathsf{B}_{h,i} := \mathsf{B}(x^o_{h,i}, \nu_2 \rho^h) \subseteq \mathsf{P}_{h,i}$,

(c) $\mathsf{B}_{h,i} \cap \mathsf{B}_{h,j} = \emptyset$

(d) For all $x \in \mathsf{X}$, $f^* - f(x) \leq \ell(x^*, x)$.

These assumptions coincide with those in (Bubeck et al., 2011a), except for the weaker local smoothness (Asm. 4.d), where the function is assumed to be Lipschitz between any two arms x, x' *close* to the maximum x^* (i.e., $|f(x) - f(x')| \leq \ell(x, x')$), while here we only require the function to be Lipschitz w.r.t. the maximum. Finally, we characterize the *complexity* of the problem using the near-optimality dimension, which defines how *large* is the set of ϵ-optimal arms in X. For the sake of clarity, we consider a slightly simplified definition of near-optimality dimension w.r.t. (Bubeck et al., 2011a).

Assumption 5 (Near-optimality dimension)**.** *Let $\epsilon = 3\nu_1\rho^h$ and $\epsilon' = \nu_2\rho^h < \epsilon$, for any subset of ϵ-optimal nodes $\mathsf{X}_\epsilon = \{x \in \mathsf{X} : f^* - f(x) \leq \epsilon\}$, there exists a constant C such that $\mathsf{N}\left(\mathsf{X}_\epsilon, \ell, \epsilon'\right) \leq C(\epsilon')^{-d}$, where d is the near-optimality dimension of f and $\mathsf{N}\left(\mathsf{X}_\epsilon, \ell, \epsilon'\right)$ is the ϵ'-cover number of X_ϵ w.r.t. the dissimilarity measure ℓ.*

Algorithm 1 The *HCT* algorithm.

Require: Parameters $\nu_1 > 0$, $\rho \in (0, 1)$, $c > 0$, tree structure $(\mathcal{P}_{h,i})_{h \geq 0, 1 \leq i \leq 2^i}$ and confidence δ.
Initialize $t = 1$, $\mathcal{T}_t = \{(0, 1), (1, 1), (1, 2)\}$, $H(t) = 1$, $U_{1,1}(t) = U_{1,2}(t) = +\infty$,
loop
  if $t = t^+$ **then** ▷ Refresh phase
    for all $(h, i) \in \mathcal{T}_t$ **do**
      $U_{h,i}(t) \leftarrow \widehat{\mu}_{h,i}(t) + \nu_1\rho^h + \sqrt{\frac{c^2 \log(1/\tilde{\delta}(t^+))}{T_{h,i}(t)}}$
    end for;
    for all $(h, i) \in \mathcal{T}_t$ Backward from $H(t)$ **do**
      if $(h, i) \in$ leaf$(\mathcal{T}_t)$ **then**
        $B_{h,i}(t) \leftarrow U_{h,i}(t)$
      else
        $B_{h,i}(t) \leftarrow \min\left[U_{h,i}(t), \max_{j \in \{2i-1, 2i\}} B_{h+1,j}(t)\right]$
      end if
    end for
  end if;
  $\{(h_t, i_t), P_t\} \leftarrow$ OptTraverse$(\mathcal{T}_t)$
  if Algorithm ***HCT*-iid then**
    Pull arm x_{h_t,i_t} and observe r_t
    $t = t + 1$
  else if Algorithm ***HCT*-Γ then**
    $T_{cur} = T_{h_t,i_t}(t)$
    while $T_{h_t,i_t}(t) < 2T_{cur}$ **AND** $t < t^+$ **do**
      Pull arm $x_{h,i}$ and observe r_t
      $(h_{t+1}, i_{t+1}) = (h_t, i_t)$
      $t = t + 1$
    end while
  end if
  Update counter $T_{h_t,i_t}(t)$ and empirical average $\widehat{\mu}_{h_t,i_t}(t)$
  $U_{h_t,i_t}(t) \leftarrow \widehat{\mu}_{h_t,i_t}(t) + \nu_1\rho^h + \sqrt{\frac{c^2 \log(1/\tilde{\delta}(t^+))}{T_{h_t,i_t}(t)}}$
  UpdateB$(\mathcal{T}_t, P_t, (h_t, i_t))$
  $\tau_h(t) = \frac{c^2 \log(1/\tilde{\delta}(t^+))}{\nu_1^2} \rho^{-2h_t}$
  if $T_{h_t,i_t}(t) \geq \tau_{h_t}(t)$ AND $(h_t, i_t) =$leaf$(\mathcal{T})$ **then**
    $\mathcal{I}_t = \{(h_t + 1, 2i_t - 1), (h_t + 1, 2i_t)\}$
    $\mathcal{T} \leftarrow \mathcal{T} \cup \mathcal{I}_t$
    $U_{h_t+1,2i_t-1}(t) = U_{h_t+1,2i_t}(t) = +\infty$
  end if
end loop

3. The High Confidence Tree algorithm

We now introduce the High Confidence Tree (HCT) algorithm. Throughout this discussion, a function evaluation corresponds to the reward received from pulling an arm. We first describe the general structure of HCT, before discussing two particular variants: *HCT*-iid, designed for the case when arm rewards are iid, and *HCT*-Γ which handles the correlated feedback case, where the reward from pulling an arm may depend on all prior arms pulled and resulting outcomes. Alg. 1 shows the structure of the algorithm for *HCT*-iid and *HCT*-Γ and their minor differences.

The general structure. The *HCT* algorithm relies on a binary covering tree T provided as input to construct a hi-

Algorithm 2 The *OptTraverse* function.

Require: Tree $\mathcal{T}$
  $(h,i) \leftarrow (0,1)$, $P \leftarrow (0,1)$
  $T_{0,1} = \tau_0(t) = 1$;
  while $(h,i) \notin \text{Leaf}(\mathcal{T})$ **AND** $T_{h,i}(t) \geq \tau_h(t)$ **do**
    if $B_{h+1,2i-1} \geq B_{h+1,2i}$ **then**
      $(h,i) \leftarrow (h+1, 2i-1)$
    else
      $(h,i) \leftarrow (h+1, 2i)$
    end if
    $P \leftarrow P \cup \{(h,i)\}$
  end while
  return (h,i) and P

Algorithm 3 The *UpdateB* function.

Require: Tree $\mathcal{T}$, the path P_t, selected node (h_t, i_t)
  if $(h_t, i_t) \in \text{Leaf}(\mathcal{T})$ **then**
    $B_{h_t,i_t}(t) = U_{h_t,i_t}(t)$
  else
    $B_{h_t,i_t}(t) = \min\left[U_{h_t,i_t}(t), \max_{j \in \{2i_t-1, 2i_t\}} B_{h_t+1,j}(t)\right]$
  end if;
  for all $(h,i) \in P_t - (h_t, i_t)$ backward **do**
    $B_{h,i}(t) = \min\left[U_{h,i}(t), \max_{j \in \{2i-1, 2i\}} B_{h+1,j}(t)\right]$
  end for

erarchical approximation of the mean-reward function f. At each node (h,i) of the tree, the algorithm keeps track of some statistics regarding the arm $x_{h,i}$ corresponding to node (h,i). These include the empirical estimate $\widehat{\mu}_{h,i}(t)$ of the mean-reward of $x_{h,i}$ computed as

$$\widehat{\mu}_{h,i}(t) := \frac{1}{T_{h,i}(t)} \sum_{s=1}^{T_{h,i}(t)} r^s(x_{h,i}), \tag{1}$$

where $T_{h,i}(t)$ is the number of times node (h,i) has been selected in the past and $r^s(x_{h,i})$ denotes the s-th reward observed after pulling $x_{h,i}$ (while we previously used r_t to denote the t-th sample of the overall process). As explained in Sect. 2, although a node is associated to a single arm $x_{h,i}$, it also covers a full portion of the input space X, i.e., the subset $\mathsf{P}_{h,i}$. Thus, similar to the HOO algorithm (Bubeck et al., 2011a), *HCT* also maintains two upper-bounds, $U_{h,i}$ and $B_{h,i}$, which are meant to bound the mean-reward $f(x)$ of all the arms $x \sqsubset \mathsf{P}_{h,i}$. For any node (h,i), the upper-bound $U_{h,i}$ is computed as

$$U_{h,i}(t) := \widehat{\mu}_{h,i}(t) + \nu_1 \rho^h + \sqrt{c^2 \frac{\log(1/\tilde{\delta}(t^+))}{T_{h,i}(t)}}, \tag{2}$$

where $t^+ = 2^{\lfloor \log(t) \rfloor + 1}$ and $\tilde{\delta}(t) := \min\{c_1 \delta / t, 1\}$. Intuitively speaking, the second term is related to the *resolution* of node (h,i) and the third term accounts for the *uncertainty* of $\widehat{\mu}_{h,i}(t)$ in estimating $f(x_{h,i})$. The B-values are designed to have a tighter upper bound on $f(x)$ by taking the minimum between $U_{h,i}$ for the current node, and the maximum upper bound of the node's two child nodes, if present.[1] More precisely,

$$B_{h,i}(t) = \begin{cases} U_{h,i}(t) & (h,i) \in \text{leaf}(\mathcal{T}_t) \\ \min[U_{h,i}(t), \max_{j \in \{2i-1, 2i\}} B_{h+1,j}(t)] & \text{otherwise.} \end{cases} \tag{3}$$

To identify which arm to pull, the algorithm traverses the tree along a path P_t obtained by selecting nodes with maximum $B_{h,i}$ until it reaches an optimistic node (h_t, i_t), which is either a leaf or a node which is not pulled enough w.r.t. to a given threshold $\tau_h(t)$, i.e., $T_{h,i}(t) \leq \tau_h(t)$ (see function *OptTraverse* in Alg. 2). Then the arm $x_{h_t,i_t} \sqsubset \mathsf{P}_{h_t,i_t}$ corresponding to selected node (h_t, i_t) is pulled.

The key element of *HCT* is the condition to decide when to expand the tree. We expand a leaf node only if we have pulled its corresponding arm a sufficient number of times such that the uncertainty over the maximum value of the arms contained within that node is dominated by the size of the subset of X it covers. Recall from Eq. 2 that the upper bound $U_{h,i}$ is composed of two terms beside the empirical average reward. The first ($\nu_1 \rho^h$) is a constant that depends only on the node depth and from assumptions 3 and 4 it follows that it bounds the possible difference in the mean-reward function between the representative arm for this node and all other arms also contained in this node, i.e., the difference between $f(x_{h,i})$ and $f(x)$ for any other $x \sqsubset \mathsf{P}_{h,i}$. The second term depends only on t and decreases with the number of pulls. At some point, the second term will become smaller than the first term, implying that the uncertainty over the rewards in $\mathsf{P}_{h,i}$ becomes dominated by the potential difference in the mean-reward of the arms in the node. This means that the domain $\mathsf{P}_{h,i}$ is too large, and thus the resolution of the current approximation of f in that region needs to be increased. Therefore *HCT* waits until these two terms become of the same magnitude before expanding a node. This happens when the number of pulls $T_{h_t,i_t}(t)$ exceeds a threshold

$$\tau_h(t) := c^2 \frac{\log(1/\tilde{\delta}(t^+))\rho^{-2h_t}}{\nu_1^2}. \tag{4}$$

(see Sect. A of the supplement for further discussion). It is at this point that expanding the node to two children can increase the accuracy of the approximation of $f(x)$, since $\nu_1 \rho^{h+1} \leq \nu_1 \rho^h$. Therefore if $T_{h_t,i_t}(t) \geq \tau_h(t)$, the algorithm expands the leaf, creates both children leaves, and set their U-values to $+\infty$. Furthermore, notice that this expansion only occurs for nodes which are likely to contain x^*. In fact, *OptTraverse* does select nodes with big B-value, which in turn receive more pulls and are thus expanded

[1] Since the node's children together contain the same input space as the node (i.e., $\mathcal{P}_{h,i} = \mathcal{P}_{h+1,2i-1} \cup \mathcal{P}_{h,2i}$), the node's maximum cannot be greater than the maximum of its children.

first. The selected arm x_{h_t,i_t} is pulled either for a single time step (in *HCT*-iid) or for a full episode (in *HCT*-Γ), and then the statistics of all the nodes along the optimistic path P_t are updated backwards. The statistics of all the nodes outside the optimistic path remain unchanged.

As *HCT* is an anytime algorithm, we periodically need to recalculate the node upper bounds to guarantee their validity with *enough* probability (see supplementary material for a more precise discussion). To do so, at the beginning of each step t, the algorithm verifies whether the B and U values need to be refreshed or not. In fact, in the definition of U in Eq. 2, the uncertainty term depends on the confidence $\tilde{\delta}(t^+)$, which changes at $t = 1, 2, 4, 8, \ldots$. Refreshing the U and B values triggers a "resampling phase" of the internal nodes of the tree T_t along the optimistic path. In fact, the second condition in the *OptTraverse* function (Alg. 2) forces *HCT* to pull arms that belong to the current optimistic path P_t until the number of pulls $T_{h,i}(t)$ becomes greater than $\tau_h(t)$ again. Notice that the choice of the confidence term $\tilde{\delta}$ is particularly critical. For instance, choosing a more natural $\tilde{\delta}(t)$ would tend to trigger the refresh (and the resampling) phase too often thus increasing the computational complexity of the algorithm and seriously affecting its theoretical properties. On the other hand, the choice of $\tilde{\delta}(t^+)$ limits the need to refresh the U and B values to only $O(\log(n))$ times over n rounds and guarantees that U and B are valid upper bounds with high probability.

***HCT*-iid and *HCT*-Γ.** The main difference between the two implementations of *HCT* is that, while *HCT*-iid pulls the selected arm for only one step before re-traversing the tree from the root to again find another optimistic node, *HCT*-Γ pulls the the representative arm of the optimistic node for an episode of T_{cur} steps, where T_{cur} is the number of pulls of arm $x_{h,i}$ at the beginning of episode. In other words, the algorithm doubles the number of pulls of each arm throughout the episode. Notice that a similar approach has been used before in other methods working with ergodic processes, such as the UCRL algorithm for ergodic MDPs (Jaksch et al., 2010). The additional stopping condition in the loop is such that not all the episodes may actually finish after T_{cur} steps and double the number of pulls: The algorithm may interrupt the episode when the confidence bounds of B and U are not valid anymore (i.e., $t \geq t^+$) and perform a refresh phase. The reason for this change is that in order to accurately estimate the mean-reward given correlated bandit feedback, it is necessary to pull an arm for a series of pulls rather than a single pull. Due to our assumption on the mixing time (Asm. 2), pulling an arm for a sufficiently long consecutive number of steps will provide an accurate estimate of the mean-reward even in the correlated setting, thus ensuring that the empirical average $\widehat{\mu}_{h,i}$ actually concentrates towards their mean value (see Lem. 2). It is this mechanism, coupled with only expanding the nodes after obtaining a good estimate of their mean reward, that allows us to handle the correlated feedback setting. Although in this sense *HCT*-Γ is more general, we do however include the *HCT*-iid variant because whenever the rewards are iid it performs better than *HCT*-Γ. This is due to the fact that, unlike *HCT*-iid, *HCT*-Γ has to keep pulling an arm for a full episode even when there is evidence that another arm could be better. We also notice that there is a small difference in the constants c_1 and c: in *HCT*-iid $c_1 := \sqrt[8]{\rho/(3\nu_1)}$ and $c := 2\sqrt{1/(1-\rho)}$, whereas *HCT*-Γ uses $c_1 := \sqrt[9]{\rho/(4\nu_1)}$ and $c := 3(3\Gamma+1)\sqrt{1/(1-\rho)}$.

4. Theoretical Analysis

In this section we analyze the regret and the complexity of *HCT*. All the proofs are reported in the supplement.

4.1. Regret Analysis

We start by reporting a bound on the maximum depth of the trees generated by *HCT*.

Lemma 1. *Given the threshold $\tau_h(t)$ in Eq. 4, the depth $H(n)$ of the tree T_n is bounded as*

$$H(n) \leq H_{\max}(n) = 1/(1-\rho)\log(n\nu_1^2/(2(c\rho)^2)). \quad (5)$$

This bound guarantees that *HCT* never expands trees beyond depth $O(\log n)$. This is ensured by the fact the *HCT* waits until the mean-reward of a node is sufficiently well estimated before expanding it and this implies that the number of pulls exponentially grows with the depth of tree, thus preventing the depth to grow linearly as in HOO.

We report regret bounds in high probability, bounds in expectation can be obtained using standard techniques.

Theorem 1 (Regret bound of *HCT*-iid)**.** *Let Assumptions 3–5 hold and at each step t, the reward r_t is independent of all prior random events. Then the regret of HCT-iid in n steps is, with probability $1-\delta$,*[2]

$$R_n \leq O\big(\big(\log(n/\delta)\big)^{1/(d+2)} n^{(d+1)/(d+2)}\big).$$

Remark (the bound). We notice that the bound perfectly matches the bound for HOO up to constants (see Thm. 6 in (Bubeck et al., 2011a)). This represents a first sanity check w.r.t. the structure of *HCT*, since it shows that changing the structure of HOO and expanding nodes only when they are pulled enough, preserves the regret properties of the algorithm. Furthermore, this result holds under milder assumptions than HOO. In fact, Asm. 4-(d) only requires f to be Lipschitz w.r.t. to the maximum x^*. Other advantages of *HCT*-iid are discussed in the Sect. 4.2 and 6.

[2]Constants are provided in Sect. A of the supplement.

Although the proof is mostly based on standard techniques and tools from bandit literature, *HCT* has a different structure from HOO (and similar algorithms) and moving from iid to correlated arms calls for the development of a significantly different proof technique. The main technical issue is to show that the empirical average $\widehat{\mu}_{h,i}$ computed by averaging rewards obtained across different episodes actually converges to $f(x_{h,i})$. In particular, we prove the following high-probability concentration inequality (see Lem. 6 in the supplement for further details).

Lemma 2. *Under Assumptions 1 and 2, for any fixed node (h,i) and step t, we have that, w.p. $1-\delta$,*

$$|\widehat{\mu}_{h,i}(t) - f(x_{h,i})| \leq (3\Gamma+1)\sqrt{\frac{2\log(5/\delta)}{T_{h,i}(t)}} + \frac{\Gamma\log(t)}{T_{h,i}(t)}.$$

Furthermore $K_{h,i}(t)$, the number of episodes in which (h,i) is selected, is bounded by $\log_2(4T_{h,i}(t)) + \log_2(t)$.

This technical lemma is at the basis of the derivation of the following regret bound for *HCT*-Γ.

Theorem 2 (Regret bound of *HCT*-Γ)**.** *Let Assumptions 1–5 hold and that rewards are generated according to the general model defined in Sect. 2. Then the regret of HCT-iid after n steps is, w.p. $1-\delta$,*

$$R_n \leq O\Big(\big(\log(n/\delta)\big)^{1/(d+2)} n^{(d+1)/(d+2)}\Big).$$

Remark (the bound). The most interesting aspect of this bound is that *HCT*-Γ achieves the same regret as *HCT*-iid when samples are non-iid. This represents a major step forward w.r.t. the existing algorithms, since it shows that the very general case of correlated rewards can be managed as well as the simpler iid case. In Sect. 5 we discuss how this result can be used in policy search for MDPs.

4.2. Complexity

Time complexity. The run time complexity of both versions of *HCT* is $O(n\log(n))$. This is due to the boundedness of the depth $H(n)$ and by the structure of the refresh phase. By Lem. 1, we have that the maximum depth is $O(\log(n))$. As a result, at each step t, the cost of traversing the tree to select a node is at most $O(\log n)$, which also coincides with the cost of updating the B and U values of the nodes in the optimistic path P_t. Thus, the total cost of selecting, pulling, and updating nodes is no larger than $O(n\log n)$. Notice that in case of *HCT*-Γ, once a node is selected is pulled for an entire episode, which further reduces the total selection cost. Another computational cost is represented by the refresh phase where all the nodes in the tree are actually updated. Since the refresh is performed only when $t = t^+$, then the number of times all the nodes are refreshed is of order of $O(\log n)$ and the boundedness of the depth guarantees that the number of nodes to update cannot be larger than $O(2^{\log n})$, which still corresponds to a total cost of $O(n\log n)$. This implies that *HCT* achieves the same run time as *T-HOO* (Bubeck et al., 2011a). Though unlike *T-HOO*, our algorithm is fully anytime and it does not suffer from the extra regret incurred due to the truncation and the doubling trick.

Space complexity. The following theorem provides bound on space complexity of the *HCT* algorithm.

Theorem 3. *Under the same conditions of Thm. 2, let N_n denote the space complexity of HCT-Γ, then we have that*

$$\mathbb{E}(\mathsf{N}_n) = O(\log(n)^{2/(d+2)} n^{d/(d+2)}).$$

This result guarantees that the space complexity of *HCT*-Γ scales sub-linearly w.r.t. n. An important observation is that the space complexity of *HCT*-Γ increases slower, by a factor of $\widetilde{O}(n^{1/(d+2)})$, than its regret. This implies that, for small values of d, HCT does not require to use a large memory space to achieve a good performance. An interesting special case is the class of problem with near-optimality dimension $d = 0$. For this class of problems the bound translates to a space complexity of $O(\log(n))$, whereas the space complexity of alternative algorithms may be as large as n (see e.g., HOO). The fact that *HCT*-Γ solves the optimization problem using only a relatively small memory space makes it a suitable choice for *big-data* applications, where the algorithms with linear space complexity can not be used due to very large size of the dataset.

Switching frequency. Finally, we also remark another interesting feature of *HCT*-Γ. Since an arm is pulled for an entire episode before another arm could be selected, this drastically reduces the number of switches between arms. In many applications, notably in reinforcement learning (see next section), this can be a significant advantage since pulling an arm may correspond to the actual implementation of a complex solution (e.g., a position in a portfolio management problem) and continuously switch between different arms might not be feasible. More formally, since each node has a number of episodes bounded by $O(\log n)$ (Lem. 2), then the number of switches can be derived from the number of nodes in Thm. 3 multiplied by $O(\log n)$, which leads to $O(\log(n)^{(d+4)/(d+2)} n^{d/(d+2)})$.

5. Application to Policy Search in MDPs

As discussed in Sect. 2, *HCT* is designed to handle the very general case of optimization in problems with strong correlation among the rewards, arm pulls, and contexts, at different time steps. An important subset of this general class is represented by the problem of policy search in infinite-horizon *ergodic* Markov decision processes.

A MDP M is defined as a tuple $(\mathsf{S}, \mathsf{A}, P)$ where S is the set of states, A is the set of actions, $P : \mathsf{S}\times\mathsf{A} \to \mathsf{M}(\mathsf{S}\times[0,1])$

is the transition kernel mapping each state-action pair to a distribution over states and rewards. A (stochastic) policy $\pi : S \to M(A)$ is a mapping from states to distribution over actions. Policy search algorithms (Scherrer & Geist, 2013; Azar et al., 2013; Kober & Peters, 2011) aim at finding the policy in a given policy set which maximizes the long-term performance. Formally, a policy search algorithm receives as input a set of policies $G = \{\pi_\theta; \theta \in \Theta\}$, each of them parameterized by a parameter vector θ in a given set $\Theta \subseteq \mathbb{R}^d$. Any policy $\pi_\theta \in G$ induces a state-reward transition kernel $T : S \times \Theta \to M(S \times [0,1])$. T relates to the state-reward-action transition kernel P and the policy kernel π_θ as follows $T(ds', dr|s, \theta) := \int_{u \in \mathcal{A}} P(ds', dr|s, u)\pi_\theta(du|s)$. For any $\pi_\theta \in G$ and initial state $s_0 \in S$, the time-average reward over n steps is $\mu^{\pi_\theta}(s_0, n) := 1/n\mathbb{E}[\sum_{t=1}^{n} r_t]$, where $r_1, r_2, \ldots, r_n$ is the sequence of rewards observed by running π_θ for n steps staring at s_0. If the Markov reward process induced by π_θ is ergodic, $\mu^{\pi_\theta}(s_0, n)$ converges to a fixed point independent of the initial state s_0. The average reward of π_θ is thus defined as $\mu(\theta) := \lim_{n\to\infty} \mu^{\pi_\theta}(s_0, n)$. The goal of policy search is to find the best $\theta^* = \arg\max_{\theta\in\Theta} \mu(\theta)$.[3]

It is straightforward now to match the MDP scenario to the general setting in Sect. 2, notably mapping Θ to X and $\mu(\theta)$ to $f(x)$ (further details are provided in Sect. D of the supplement). This allows us to directly apply *HCT*-Γ to the problem of policy search. The advantage of *HCT*-Γ algorithm w.r.t. prior work is that, to the best of our knowledge, it is the first policy search algorithm which provides finite sample guarantees in the form of regret bounds on the performance loss of policy search in MDPs (see Thm. 2), which guarantee that *HCT*-Γ suffers from a small sub-linear regret w.r.t. π_{θ^*}. Also, it is possible to prove that the policy induced by *HCT*-Γ has a small simple regret, that is, the average reward of the policy chosen by *HCT*-Γ converges to $\mu(\theta^*)$ with a polynomial rate.[4] Another interesting feature of *HCT*-Γ is that it can be used in large (continuous) state-action problems since it does not make any restrictive assumption on the size of state-action space.

Related work. A related work to *HCT*-Γ is the UCCRL algorithm by Ortner & Ryabko (2012), which extends the original UCRL algorithm (Jaksch et al., 2010) to continuous state spaces. Although a direct comparison between the two methods is not possible, it is interesting to notice that the assumptions used in UCCRL are stronger than for *HCT*-Γ, since they require both the dynamics and the reward function to be globally Lipschitz. Furthermore, UCCRL requires the action space to be finite, while *HCT*-Γ can deal with any continuous policy space. Finally, while *HCT*-Γ minimizes the regret against the best policy in G, UCCRL targets the performance of the actual optimal policy of the MDP at hand. Another relevant work is the OMDP algorithm of Abbasi et al. (2013) which deals with the problem of RL in continuous state-action MDPs with adversarial rewards. OMDP achieves a sub-linear regret under the assumption that the space of policies is finite.

[3]Note that π_{θ^*} is optimal in the policy class $\mathcal{G}$ and it may not coincide with the optimal policy π^* of the MDP.

[4]Refer to Bubeck et al. (2011a); Munos (2013) for how to transform cumulative regret bounds to simple regret bounds.

6. Numerical Results

In this section we provide preliminary simulation results to demonstrate some properties of *HCT*.

Setup. We focus on minimizing the regret across repeated noisy evaluations of the garland function $f(x) = x(1-x)(4-\sqrt{|\sin(60x)|})$ relative to repeatedly selecting its global optima.[5] We evaluate the performance of each algorithm in terms of the per-step regret, $\widetilde{R}_n = R_n/n$. Each run is $n = 10^5$ steps and we average the performance on 10 runs. For all the algorithms compared in the following, parameters[6] are optimized to maximize their performance.

I.i.d. setting. In the first experiment we compare *HCT*-iid to the truncated hierarchical optimistic optimization (T-HOO) algorithm (Bubeck et al., 2011a). T-HOO is a state-of-the-art X-armed bandit algorithm, developed as a computationally-efficient alternative of HOO. In Fig. 1 we show the per-step regret, the runtime, and the space requirements of each approach. As predicted by the theoretical bounds, the per-step regret $\widetilde{R}_n$ of both *HCT*-iid and *T-HOO* decreases rapidly with number of steps. Though the big-O bounds are identical for both approaches, empirically we observe that in this setting *HCT*-iid outperforms *T-HOO* by a large margin. Similarly, though the computational complexity of both approaches matches in the dependence on the number of time steps, empirically we observe that our approach outperforms *T-HOO* (Fig. 1). Perhaps the most significant expected advantage of HCT-iid over T-HOO for iid settings is in the space requirements. HCT-iid has a space requirement for this domain that scales logarithmically with the time step n, as predicted by Thm. 3. In contrast, in this domain we observe a polynomial growth of memory usage for T-HOO. These patterns mean that *HCT*-iid can achieve a very small regret using a sparse cover tree with only few hundred nodes, whereas *T-HOO* requires orders of magnitude more nodes than *HCT*-iid.

Correlated setting. In this setting, we compare *HCT*-Γ to PoWER, a standard RL policy search algorithm (Kober & Peters, 2011), on a continuous-state-action MDP constructed out of the garland function.[7] PoWER uses an

[5]We discuss the properties of the garland function in Sect. C.

[6]For both HCT and T-HOO we introduce a tuning parameter used to multiply the upper bounds, while for PoWER we optimize the window for computing the weighted average.

[7]See Sect. C of the supplement for details.

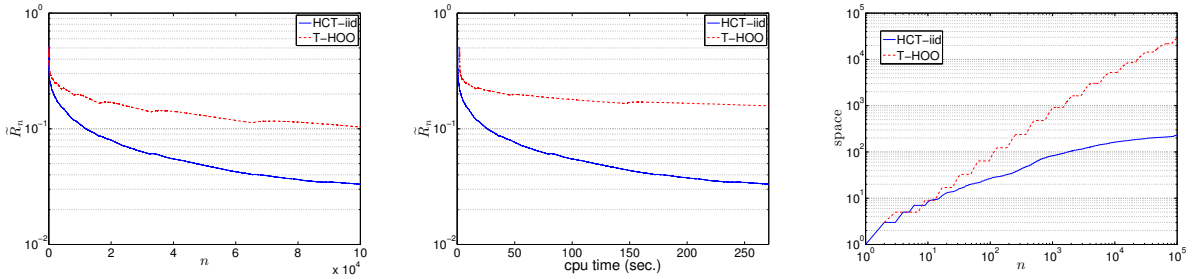

Figure 1. Comparison of the Performance of HCT-iid and the Previous Methods under the iid Bandit Feedback.

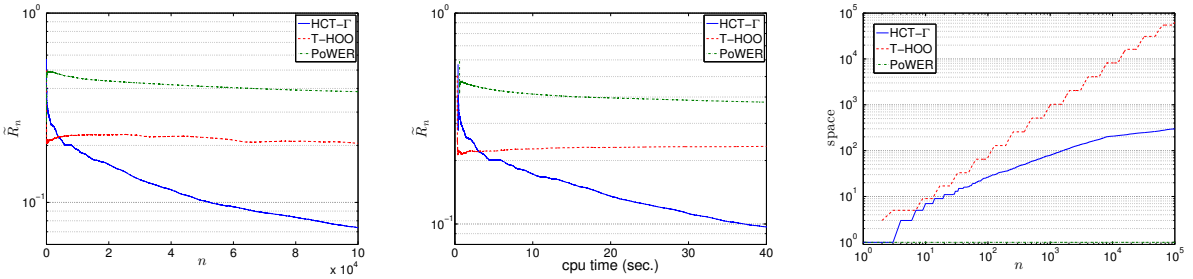

Figure 2. Comparison of the Performance of HCT $-$ Γ and the Previous Methods under Correlated Bandit Feedback (MDP setting)

Expectation Maximization approach to optimize the policy parameters. We also compare our algorithm with T-HOO, although this algorithm is designed for the iid setting and it may fail to converge to the global optimum under correlated bandit feedback. Fig. 2 shows per-step regret of the 3 approaches in the MDP. Only *HCT*-Γ succeeds in finding the globally optimal policy, since only for *HCT*-Γ the average regret tends to converge to zero (as predicted by Thm. 2). The PoWER method finds worse solutions than both stochastic optimization approaches for the same amount of computational time, likely due to using EM which is known to be susceptible to local optima. On the other hand, its primary advantage is that it has a very small memory requirement. Overall this illustrates the benefit of *HCT* for online MDP policy search, since it can quickly (as a function of samples and runtime) find a global optima, and is, to our knowledge, one of the only policy search methods guaranteed to do so.

7. Discussion and Future Work

In this paper we introduced a new X-armed bandit algorithm for optimization under bandit feedback and prove regret bounds and simulation results for it. Our approach improves on existing results to handle the important case of correlated bandit feedback. This allows *HCT* to be applied to a broader range of problems than prior X-armed bandit algorithms, such as policy search in continuous MDPs.

In the current version of *HCT* we assume that the learner has access to the information regarding the smoothness of function $f(x)$ and the mixing time Γ. In many problems those information are not available to the learner. In the future it would be interesting to build on prior work that handles unknown smoothness in iid settings and extend it to correlated feedback. For example, Bubeck et al. (2011b) require a stronger global Lipschitz assumption and propose an algorithm to estimate the Lipschitz constant. Other work on the iid setting include Valko et al. (2013) and Munos (2011), which are limited to the simple regret scenario, but who only use the mild local smoothness assumption we define in Asm. 4, and do not require knowledge of the dissimilarity measure ℓ. On the other hand, Slivkins (2011) and Bull (2013) study the cumulative regret but consider a different definition of smoothness related to the zooming concept introduced by Kleinberg et al. (2008). Finally, we notice that to deal with unknown mixing time, one may rely on data-dependent tail's inequalities, such as empirical Bernstein inequality (Tolstikhin & Seldin, 2013; Maurer & Pontil, 2009), replacing the mixing time with the empirical variance of the rewards. In the future we also wish to explore using *HCT* in other problems that can be modeled as optimization with correlated bandit feedback. For example, *HCT* may be used for policy search in partially observable MDPs (Vlassis & Toussaint, 2009; Baxter & Bartlett, 2000), as long as the POMDP is ergodic.

Acknowledgements

This work was supported in part by the NSF Award SBE-0836012 to the Pittsburgh Sciences of Learning Center. A. Lazaric acknowledges the support of the Ministry of Higher Education and Research, Nord-Pas-de-Calais Regional Council and FEDER through the Contrat de Projets Etat Region (CPER) 2007-2013, and the European Communitys Seventh Framework Programme (FP7/2007-2013) under grant 231495 (project CompLACS).

References

Abbasi, Yasin, Bartlett, Peter, Kanade, Varun, Seldin, Yevgeny, and Szepesvari, Csaba. Online learning in markov decision processes with adversarially chosen transition probability distributions. In Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K.Q. (eds.), *Advances in Neural Information Processing Systems 26*, pp. 2508–2516. 2013.

Auer, Peter, Ortner, Ronald, and Szepesvári, Csaba. Improved rates for the stochastic continuum-armed bandit problem. In *COLT*, pp. 454–468, 2007.

Azar, Mohammad Gheshlaghi, Lazaric, Alessandro, and Brunskill, Emma. Regret bounds for reinforcement learning with policy advice. In *ECML/PKDD*, pp. 97–112, 2013.

Baxter, Jonathan and Bartlett, Peter L. Reinforcement learning in pomdp's via direct gradient ascent. In *ICML*, pp. 41–48, 2000.

Bubeck, Sébastien, Munos, Rémi, Stoltz, Gilles, and Szepesvári, Csaba. *X*-armed bandits. *Journal of Machine Learning Research*, 12:1655–1695, 2011a.

Bubeck, Sébastien, Stoltz, Gilles, and Yu, Jia Yuan. Lipschitz bandits without the lipschitz constant. In *ALT*, pp. 144–158, 2011b.

Bull, Adam. Adaptive-tree bandits. *arXiv preprint arXiv:1302.2489*, 2013.

Cope, Eric. Regret and convergence bounds for immediate-reward reinforcement learning with continuous action spaces. *IEEE Transactions on Automatic Control*, 54(6): 1243–1253, 2009.

Djolonga, Josip, Krause, Andreas, and Cevher, Volkan. High dimensional gaussian process bandits. In *Neural Information Processing Systems (NIPS)*, 2013.

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

Kleinberg, Robert, Slivkins, Aleksandrs, and Upfal, Eli. Multi-armed bandits in metric spaces. In *STOC*, pp. 681–690, 2008.

Kleinberg, Robert, Slivkins, Aleksandrs, and Upfal, Eli. Bandits and experts in metric spaces. *arXiv preprint arXiv:1312.1277*, 2013.

Kober, Jens and Peters, Jan. Policy search for motor primitives in robotics. *Machine Learning*, 84(1-2):171–203, 2011.

Lattimore, Tor, Hutter, Marcus, and Sunehag, Peter. The sample-complexity of general reinforcement learning. In *Proceedings of Thirtieth International Conference on Machine Learning (ICML)*, 2013.

Levin, David A., Peres, Yuval, and Wilmer, Elizabeth L. *Markov chains and mixing times*. American Mathematical Society, 2006.

Maurer, Andreas and Pontil, Massimiliano. Empirical bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*, 2009.

Munos, Rémi. Optimistic optimization of a deterministic function without the knowledge of its smoothness. In *NIPS*, pp. 783–791, 2011.

Munos, Rémi. From bandits to monte-carlo tree search: The optimistic principle applied to optimization and planning. *Foundations and Trends in Machine Learning*, 2013.

Ortner, Ronald and Ryabko, Daniil. Online regret bounds for undiscounted continuous reinforcement learning. In Bartlett, P., Pereira, F.c.n., Burges, C.j.c., Bottou, L., and Weinberger, K.q. (eds.), *Advances in Neural Information Processing Systems 25*, pp. 1772–1780, 2012.

Scherrer, Bruno and Geist, Matthieu. Policy search: Any local optimum enjoys a global performance guarantee. *arXiv preprint arXiv:1306.1520*, 2013.

Slivkins, Aleksandrs. Contextual bandits with similarity information. *CoRR*, abs/0907.3986, 2009.

Slivkins, Aleksandrs. Multi-armed bandits on implicit metric spaces. In *Advances in Neural Information Processing Systems*, pp. 1602–1610, 2011.

Srinivas, Niranjan, Krause, Andreas, Kakade, Sham M., and Seeger, Matthias. Gaussian process bandits without regret: An experimental design approach. *CoRR*, abs/0912.3995, 2009.

Tolstikhin, Ilya O and Seldin, Yevgeny. PAC-bayes-empirical-bernstein inequality. In *Advances in Neural Information Processing Systems*, pp. 109–117, 2013.

Valko, Michal, Carpentier, Alexandra, and Munos, Rémi. Stochastic simultaneous optimistic optimization. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pp. 19–27, 2013.

Vlassis, Nikos and Toussaint, Marc. Model-free reinforcement learning as mixture learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 1081–1088, 2009.

A. Proof of Thm. 1

In this section we report the full proof of the regret bound of *HCT*-iid.

We begin by introducing some additional notation, required for the analysis of both algorithms. We denote the indicator function of an event $\mathcal{E}$ by $\mathbb{I}_{\mathcal{E}}$. For all $1 \leq h \leq H(t)$ and $t > 0$, we denote by $\mathcal{I}_h(t)$ the set of all nodes created by the algorithm at depth h up to time t and by $\mathcal{I}_h^+(t)$ the subset of $\mathcal{I}_h(t)$ including only the internal nodes (i.e., nodes that are not leaves), which corresponds to nodes at depth h which have been expanded before time t. At each time step t, we denote by (h_t, i_t) the node selected by the algorithm. For every $(h,i) \in \mathcal{T}$, we define the set of time steps when (h,i) has been selected as $\mathcal{C}_{h,i} := \{t = 1, \dots, n : (h_t, i_t) = (h,i)\}$. We also define the set of times that a child of (h,i) has been selected as $\mathcal{C}^c_{h,i} := \mathcal{C}_{h+1,2i-1} \bigcup \mathcal{C}_{h+1,2i}$. We need to introduce three important steps related to node (h,i):

- $\bar{t}_{h,i} := \max_{t \in \mathcal{C}_{h,i}} t$ is the last time (h,i) has been selected,
- $\tilde{t}_{h,i} := \max_{t \in \mathcal{C}^c_{h,i}} t$ is the last time when any of the two children of (h,i) has been selected,
- $t_{h,i} := \min\{t : T_{h,i}(t) > \tau_h(t)\}$ is the step when (h,i) is expanded.

The choice of τ_h. The threshold on the the number of pulls needed before expanding a node at depth h is determined so that, at each time t, the two confidence terms in the definition of U (Eq. 2) are roughly equivalent, that is

$$\nu_1 \rho^h = c\sqrt{\frac{\log(1/\tilde{\delta}(t^+))}{\tau_h(t)}} \quad \Leftrightarrow \quad \tau_h(t) = \frac{c^2 \log(1/\tilde{\delta}(t^+))}{\nu_1^2} \rho^{-2h}.$$

Furthermore, since $t \leq t^+ \leq 2t$ then

$$\frac{c^2}{\nu_1^2}\rho^{-2h} \leq \frac{c^2 \log(1/\tilde{\delta}(t))}{\nu_1^2}\rho^{-2h} \leq \tau_h(t) \leq \frac{c^2 \log(2/\tilde{\delta}(t))}{\nu_1^2}\rho^{-2h}, \tag{6}$$

where we used the fact that $0 < \tilde{\delta}(t) \leq 1$ for all $t > 0$. As described in Section 3, the idea is that the expansion of a node, which corresponds to an increase in the resolution of the approximation of f, should not be performed until the empirical estimate $\widehat{\mu}_{h,i}$ of $f(x_{h,i})$ is accurate enough. Notice that the number of pulls $T_{h,i}(t)$ for an expanded node (h,i) does not necessarily coincide with $\tau_h(t)$, since t might correspond to a time step when some leaves have not been pulled until $\tau_h(t)$ and other nodes have not been fully resampled after a refresh phase.

We begin our analysis by bounding the maximum depth of the trees constructed by *HCT*-iid.

Lemma 1 *Given the number of samples $\tau_h(t)$ required for the expansion of nodes at depth h in Eq. 4, the depth $H(n)$ of the tree $\mathcal{T}_n$ is bounded as*

$$H(n) \leq H_{\max}(n) = \frac{1}{1-\rho}\log\Big(\frac{n\nu_1^2}{2(c\rho)^2}\Big).$$

Proof. The deepest tree that can be developed by *HCT*-iid is a *linear* tree, where at each depth h only one node is expanded, that is , $|\mathcal{I}_h^+(n)| = 1$ and $|\mathcal{I}_h(n)| = 2$ for all $h < H(n)$. Thus we have

$$\begin{aligned} n &= \sum_{h=0}^{H(n)} \sum_{i \in \mathcal{I}_h(n)} T_{h,i}(n) \geq \sum_{h=0}^{H(n)-1} \sum_{i \in \mathcal{I}_h(n)} T_{h,i}(n) \geq \sum_{h=0}^{H(n)-1} \sum_{i \in \mathcal{I}_h^+(n)} T_{h,i}(n) \geq \sum_{h=0}^{H(n)-1} \sum_{i \in \mathcal{I}_h^+(n)} T_{h,i}(t_{h,i}) \\ &\overset{(1)}{\geq} \sum_{h=0}^{H(n)-1} \sum_{i \in \mathcal{I}_h^+(n)} \tau_{h,i}(t_{h,i}) \geq \sum_{h=1}^{H(n)-1} \frac{c^2}{\nu_1^2}\rho^{-2h} \geq \frac{(c\rho)^2}{\nu_1^2}\rho^{-2H(n)} \sum_{h=1}^{H(n)-1} \rho^{-2(h-H(n)+1)}, \end{aligned}$$

where inequality (1) follows from the fact that a node (h,i) is expanded at time $t_{h,i}$ only when it is pulled *enough*, i.e., $T_{h,i}(t_{h,i}) \geq \tau_h(t_{h,i})$. Since all the elements in the summation over h are positive, then we can lower-bound the sum by its last element ($h = H(n)$), which is 1, and obtain

$$n \geq 2\frac{(c\rho)^2}{\nu_1^2} H(n) \rho^{-2H(n)} \geq 2\frac{(c\rho)^2}{\nu_1^2}\rho^{-2H(n)},$$

where we used the fact that $H(n) \geq 1$. By solving the previous expression we obtain

$$\rho^{-2H(n)} \leq n\frac{\nu_1^2}{2(c\rho)^2} \quad \Rightarrow \quad H(n) \leq \frac{1}{2}\log\Big(\frac{n\nu_1^2}{2(c\rho)^2}\Big)/\log(1/\rho).$$

Finally, the statement follows using $\log(1/\rho) \geq 1-\rho$. □

We now introduce a high probability event under which the mean reward for all the expanded nodes is within a confidence interval of the empirical estimates at a *fixed* time t.

Lemma 3 (High-probability event)**.** *We define the set of all the possible nodes in trees of maximum depth $H_{\max}(t)$ as*

$$\mathcal{L}_t = \bigcup_{\mathcal{T}:\mathit{Depth}(\mathcal{T})\leq H_{\max}(t)} \mathit{Nodes}(\mathcal{T}).$$

We introduce the event

$$\mathcal{E}_t = \left\{\forall (h,i) \in \mathcal{L}_t, \forall T_{h,i}(t) = 1..t : \left|\widehat{\mu}_{h,i}(t) - f(x_{h,i})\right| \leq c\sqrt{\frac{\log(1/\tilde{\delta}(t))}{T_{h,i}(t)}}\right\},$$

where $x_{h,i} \in \mathcal{P}_{h,i}$ is the arm corresponding to node (h,i). If

$$c = 2\sqrt{\frac{1}{1-\rho}} \quad \textit{and} \quad \tilde{\delta}(t) = \frac{\delta}{t}\sqrt[3]{\frac{\rho}{3\nu_1}},$$

then for any fixed t, the event $\mathcal{E}_t$ holds with probability at least $1-\delta/t^6$.

Proof. We upper bound the probability of the complementary event as

$$\begin{aligned}
\mathbb{P}[\mathcal{E}_t^c] &\leq \sum_{(h,i)\in\mathcal{L}_t}\sum_{T_{h,i}(t)=1}^{t} \mathbb{P}\left[\left|\widehat{\mu}_{h,i}(t) - \mu_{h,i}\right| \geq c\sqrt{\frac{\log(1/\tilde{\delta}(t))}{T_{h,i}(t)}}\right] \\
&\leq \sum_{(h,i)\in\mathcal{L}_t}\sum_{T_{h,i}(t)=1}^{t} 2\exp\left(-2T_{h,i}(t)c^2\frac{\log(1/\tilde{\delta}(t))}{T_{h.i}(t)}\right) \\
&= 2\exp\big(-2c^2\log(1/\tilde{\delta}(t))\big)t|\mathcal{L}_t|,
\end{aligned}$$

where the first inequality is an application of a union bound and the second inequality follows from the Chernoff-Hoeffding inequality. We upper bound the number of nodes in $\mathcal{L}_t$ by the largest binary tree with a maximum depth $H_{\max}(t)$, i.e., $|\mathcal{L}_t| \leq 2^{H_{\max}(t)+1}$. Thus

$$\mathbb{P}[\mathcal{E}_t^c] \leq 2(\tilde{\delta}(t))^{2c^2}t2^{H_{\max}(t)+1}.$$

We first derive a bound on the the term $2^{H_{\max}(t)}$ as

$$2^{H_{\max}(t)} \leq \mathrm{pow}\left(2, \log_2\left(\frac{t\nu_1^2}{2(c\rho)^2}\right)^{\frac{1}{2\log_2(e)(1-\rho)}}\right) \leq \left(\frac{t\nu_1^2}{2(c\rho)^2}\right)^{\frac{1}{2(1-\rho)}},$$

where we used the upper bound $H_{\max}(t)$ from Lemma 1 and $\log_2(e) > 1$. This leads to

$$\mathbb{P}[\mathcal{E}_t^c] \leq 4t\big(\tilde{\delta}(t)\big)^{2c^2}\left(\frac{t\nu_1^2}{2(c\rho)^2}\right)^{\frac{1}{2(1-\rho)}}.$$

The choice of c and $\tilde{\delta}(t)$ as in the statement leads to

$$\begin{aligned}
\mathbb{P}[\mathsf{E}_t^{\mathsf{c}}] &\leq 4t\left(\sqrt[8]{\rho/(3\nu_1)}\delta/t\right)^{\frac{8}{1-\rho}}\left(\frac{t\nu_1^2(1-\rho)}{8\rho^2}\right)^{\frac{1}{2(1-\rho)}} \\
&= 4t\big(\delta/t\big)^{\frac{8}{1-\rho}}\big(\rho/(3\nu_1)\big)^{\frac{1}{1-\rho}}t^{\frac{1}{2(1-\rho)}}\left(\frac{\nu_1\sqrt{1-\rho}}{\sqrt{8}\rho}\right)^{\frac{1}{1-\rho}} \\
&\leq 4\delta t^{1-\frac{8}{1-\rho}+\frac{1}{2(1-\rho)}}\left(\frac{\sqrt{1-\rho}}{3\sqrt{8}}\right)^{\frac{1}{1-\rho}} \\
&\leq \sqrt[4]{\frac{1}{3\sqrt{8}}}\delta t^{\frac{-2\rho-13}{2(1-\rho)}} \leq \delta t^{-13/2} \leq \delta/t^6,
\end{aligned}$$

which completes the proof. □

Recalling the definition the regret from Sect. s:preliminaries, we decompose the regret of *HCT*-iid in two terms depending on whether event E_t holds or not (i.e., failing confidence intervals). Let the instantaneous regret be $\Delta_t = f^* - r_t$, then we rewrite the regret as

$$R_n = \sum_{t=1}^{n}\Delta_t = \sum_{t=1}^{n}\Delta_t\mathbb{I}_{\mathcal{E}_t} + \sum_{t=1}^{n}\Delta_t\mathbb{I}_{\mathcal{E}_t^{\mathsf{c}}} = R_n^{\mathcal{E}} + R_n^{\mathcal{E}^{\mathsf{c}}}. \tag{7}$$

We first study the regret in the case of failing confidence intervals.

Lemma 4 (Failing confidence intervals)**.** *Given the parameters c and $\tilde{\delta}(t)$ as in Lemma 3, the regret of HCT-iid when confidence intervals fail to hold is bounded as*

$$R_n^{\mathcal{E}^{\mathsf{c}}} \leq \sqrt{n},$$

with probability $1 - \frac{\delta}{5n^2}$.

Proof. We first split the time horizon n in two phases: the first phase until $\sqrt{n}$ and the rest. Thus the regret becomes

$$R_n^{\mathcal{E}^{\mathsf{c}}} = \sum_{t=1}^{n}\Delta_t\mathbb{I}_{\mathcal{E}_t^{\mathsf{c}}} = \sum_{t=1}^{\sqrt{n}}\Delta_t\mathbb{I}_{\mathcal{E}_t^{\mathsf{c}}} + \sum_{t=\sqrt{n}+1}^{n}\Delta_t\mathbb{I}_{\mathcal{E}_t^{\mathsf{c}}}.$$

We trivially bound the regret of first term by $\sqrt{n}$. So in order to prove the result it suffices to show that event $\mathsf{E}_t^{\mathsf{c}}$ never happens after $\sqrt{n}$, which implies that the remaining term is zero with high probability. By summing up the probabilities $\mathbb{P}[\mathsf{E}_t^{\mathsf{c}}]$ from $\sqrt{n}+1$ to n and applying union bound we deduce

$$\mathbb{P}\left[\bigcup_{t=\sqrt{n}+1}^{n}\mathsf{E}_t^{\mathsf{c}}\right] \leq \sum_{t=\sqrt{n}+1}^{n}\mathbb{P}[\mathsf{E}_t^{\mathsf{c}}] \leq \sum_{\sqrt{n}+1}^{n}\frac{\delta}{t^6} \leq \int_{\sqrt{n}}^{+\infty}\frac{\delta}{t^6}dt \leq \frac{\delta}{5n^{5/2}} \leq \frac{\delta}{5n^2}.$$

In words this result implies that w.p. $\geq 1 - \delta/(5n^2)$ we can not have a failing confidence interval after time $\sqrt{n}$. This combined with the trivial bound of $\sqrt{n}$ for the first $\sqrt{n}$ steps completes the proof. □

We are now ready to prove the main theorem, which only requires to study the regret term under events $\{\mathsf{E}_t\}$.

Theorem 1 (Regret bound of *HCT*-iid). *Let $\delta \in (0,1)$, $\tilde{\delta}(t) = \sqrt[8]{\rho/(3\nu_1)}\delta/t$, and $c = 2\sqrt{1/(1-\rho)}$. We assume that assumptions 3–5 hold and that at each step t, the reward r_t is independent of all prior random events and $\mathbb{E}(r_t|x_t) = f(x_t)$. Then the regret of HCT-iid after n steps is*

$$R_n \leq 3\left(\frac{2^{2d+7}\nu_1^{2(d+1)}C\nu_2^{-d}\rho^d}{(1-\rho)^{d+7}}\right)^{\frac{1}{d+2}}\left(\log\left(\frac{2n}{\delta}\sqrt[8]{\frac{3\nu_1}{\rho}}\right)\right)^{\frac{1}{d+2}}n^{\frac{d+1}{d+2}} + 2\sqrt{n\log(4n/\delta)},$$

with probability $1 - \delta$.

Proof. **Step 1: Decomposition of the regret.** We start by further decomposing the regret in two terms. We rewrite the instantaneous regret Δ_t as

$$\Delta_t = f^* - r_t = f^* - f(x_{h_t,i_t}) + f(x_{h_t,i_t}) - r_t = \Delta_{h_t,i_t} + \widehat{\Delta}_t,$$

which leads to a regret (see Eq. 7)

$$R_n^{\mathcal{E}} = \sum_{t=1}^{n} \Delta_{h_t,i_t} \mathbb{I}_{\mathcal{E}_t} + \sum_{t=1}^{n} \widehat{\Delta}_t \mathbb{I}_{\mathcal{E}_t} \le \sum_{t=1}^{n} \Delta_{h_t,i_t} \mathbb{I}_{\mathcal{E}_t} + \sum_{t=1}^{n} \widehat{\Delta}_t = \widetilde{R}_n^{\mathcal{E}} + \widehat{R}_n^{\mathcal{E}}. \tag{8}$$

We start bounding the second term. We notice that the sequence $\{\widehat{\Delta}_t\}_{t=1}^n$ is a bounded martingale difference sequence since $\mathbb{E}(\widehat{\Delta}_t | \mathcal{F}_{t-1}) = 0$ and $|\widehat{\Delta}_t| \le 1$. Therefore, an immediate application of the Azuma's inequality leads to

$$\widehat{R}_n^{\mathcal{E}} = \sum_{t=1}^{n} \widehat{\Delta}_t \le 2\sqrt{n \log(4n/\delta)}, \tag{9}$$

with probability $1 - \delta/(4n^2)$.

Step 2: Preliminary bound on the regret of selected nodes and their parents. We now proceed with the study of the first term $\widetilde{R}_n^{\mathcal{E}}$, which refers to the regret of the selected nodes as measured by its mean-reward. We start by characterizing which nodes are actually selected by the algorithm under event $\mathcal{E}_t$. Let (h_t, i_t) be the node chosen at time t and P_t be the path from the root to the selected node. Let $(h', i') \in P_t$ and (h'', i'') be the node which immediately follows (h', i') in P_t (i.e., $h'' = h' + 1$). By definition of B and U values, we have that

$$B_{h',i'}(t) = \min\left[U_{h',i'}(t); \max\left(B_{h'+1,2i'-1}(t); B_{h'+1,2i'}(t)\right)\right] \le \max\left(B_{h'+1,2i'-1}(t); B_{h'+1,2i'}(t)\right) = B_{h'',i''}(t), \tag{10}$$

where the last equality follows from the fact that the *OptTraverse* function selects the node with the largest B value. By iterating the previous inequality for all the nodes in P_t until the selected node (h_t, i_t) and its parent (h_t^p, i_t^p), we obtain that

$$\begin{aligned} B_{h',i'}(t) &\le B_{h_t,i_t}(t) \le U_{h_t,i_t}(t), \qquad \forall (h', i') \in P_t \\ B_{h',i'}(t) &\le B_{h_t^p,i_t^p}(t) \le U_{h_t^p,i_t^p}(t), \qquad \forall (h', i') \in P_t - (h_t, i_t) \end{aligned}$$

by definition of B-values. Thus for any node $(h, i) \in P_t\}$, we have that $U_{h_t,i_t}(t) \ge B_{h,i}(t)$. Furthermore, since the root node $(0, 1)$ which covers the whole arm space $\mathcal{X}$ is in P_t, thus there exists at least one node (h^*, i^*) in the set P_t which includes the maximizer x^* (i.e., $x^* \in \mathcal{P}_{h^*,i^*}$) and has the the depth $h^* \le h_t^p < h_t$.[8] Thus

$$\begin{aligned} U_{h_t,i_t}(t) &\ge B_{h^*,i^*}(t). \\ U_{h_t^p,i_t^p}(t) &\ge B_{h^*,i^*}(t) \end{aligned} \tag{11}$$

Notice that in the set P_t we may have multiple nodes (h^*, i^*) which contain x^* and that for all of them we have the following sequence of inequalities holds

$$f^* - f(x_{h^*,i^*}) \le \ell(x^*, x_{h^*,i^*}) \le \operatorname{diam}(\mathcal{P}_{h^*,i^*}) \le \nu_1 \rho^{h^*}, \tag{12}$$

where the second inequality holds since $x^* \in \mathcal{P}_{h^*,i^*}$.

Now we expand the inequality in Eq. 11 on both sides using the high-probability event $\mathcal{E}_t$. First we have

$$\begin{aligned} U_{h_t,i_t}(t) &= \widehat{\mu}_{h_t,i_t}(t) + \nu_1 \rho^{h_t} + c\sqrt{\frac{\log(1/\tilde{\delta}(t^+))}{T_{h_t,i_t}(t)}} \le f(x_{h_t,i_t}) + c\sqrt{\frac{\log(1/\tilde{\delta}(t))}{T_{h_t,i_t}(t)}} + \nu_1 \rho^{h_t} + c\sqrt{\frac{\log(1/\tilde{\delta}(t^+))}{T_{h_t,i_t}(t)}} \\ &\le f(x_{h_t,i_t}) + \nu_1 \rho^{h_t} + 2c\sqrt{\frac{\log(1/\tilde{\delta}(t^+))}{T_{h_t,i_t}(t)}}, \end{aligned} \tag{13}$$

[8] Note that we never pull the root node $(0, 1)$, therefore $h_t > 0$.

where the first inequality holds on E by definition of U and the second by the fact that $t^+ \geq t$ (and $\log(1/\tilde{\delta}(t)) \leq \log(1/\tilde{\delta}(t^+))$). The same result also holds for (h_t^p, i_t^p) at time t:

$$U_{h_t^p,i_t^p}(t) \leq f(x_{h_t^p,i_t^p}) + \nu_1 \rho^{h_t^p} + 2c\sqrt{\frac{\log(1/\tilde{\delta}(t^+))}{T_{h_t^p,i_t^p}(t)}}. \tag{14}$$

We now show that for any node (h^*, i^*) such that $x^* \sqsupset \mathsf{P}_{h^*,i^*}$, then $U_{h^*,i^*}(t)$ is a valid upper bound on f^*:

$$\begin{aligned} U_{h^*,i^*}(t) &= \widehat{\mu}_{h^*,i^*}(t) + \nu_1 \rho^{h} + c\sqrt{\frac{\log(1/\tilde{\delta}(t^+))}{T_{h^*,i^*}(t)}} \overset{(1)}{\geq} \widehat{\mu}_{h^*,i^*}(t) + \nu_1 \rho^{h^*} + c\sqrt{\frac{\log(1/\tilde{\delta}(t))}{T_{h^*,i^*}(t)}} \\ &\overset{(2)}{\geq} f(x_{h^*,i^*}) + \nu_1 \rho^{h^*} \overset{(3)}{\geq} f^*, \end{aligned}$$

where (1) follows from the fact that $t^+ \geq t$, on (2) we rely on the fact that the event E_t holds at time t and on (3) we use the regularity of the function w.r.t. the maximum f^* from Eq. 12. If an optimal node (h^*, i^*) is a leaf, then $B_{h^*,i^*}(t) = U_{h^*,i^*}(t) \geq f^*$. In the case that (h^*, i^*) is not a leaf, there always exists a leaf (h^+, i^+) such that $x^* \sqsupset \mathsf{P}_{h^+,i^+}$ for which (h^*, i^*) is its ancestor, since all the optimal nodes with $h > h^*$ are descendants of (h^*, i^*). Now by propagating the bound backward from (h^+, i^+) to (h^*, i^*) through Eq. 3 (see Eq. 10) we can show that $B_{h^*,i^*}(t)$ is still a valid upper bound of the optimal value f^*. Thus for any optimal node (h^*, i^*) at time t under the event E_t we have

$$B_{h^*,i^*}(t) \geq f^*.$$

Combining this with Eq. 13, Eq. 14 and Eq. 11 , we obtain that on event E_t the selected node (h_t, i_t) and its parent (h_t^p, i_t^p) at any time t is such that

$$\begin{aligned} \Delta_{h_t,i_t} &= f^* - f(x_{h_t,i_t}) \leq \nu_1 \rho^{h_t} + 2c\sqrt{\frac{\log(1/\tilde{\delta}(t^+))}{T_{h_t,i_t}(t)}}. \\ \Delta_{h_t^p,i_t^p} &= f^* - f(x_{h_t^p,i_t^p}) \leq \nu_1 \rho^{h_t^p} + 2c\sqrt{\frac{\log(1/\tilde{\delta}(t^+))}{T_{h_t^p,i_t^p}(t)}}. \end{aligned} \tag{15}$$

Furthermore, since *HCT*-iid only selects nodes with $T_{h,i}(t) < \tau_h(t)$ the previous expression can be further simplified as

$$\Delta_{h_t,i_t} \leq 3c\sqrt{\frac{\log(2/\tilde{\delta}(t))}{T_{h_t,i_t}(t)}}, \tag{16}$$

where we also used that $t^+ \leq 2t$ for any t. Although this provides a preliminary bound on the instantaneous regret of the selected nodes, we need to further refine this bound.

In the case of parent (h_t^p, i_t^p), since $T_{h_t^p,i_t^p}(t) \geq \tau_{h_t^p}(t)$, we deduce

$$\Delta_{h_t^p,i_t^p} \leq \nu_1 \rho^{h_t^p} + 2c\sqrt{\frac{\log(1/\tilde{\delta}(t^+))}{\tau_{h_t^p}(t)}} = 3\nu_1 \rho^{h_t^p}, \tag{17}$$

This implies that every selected node (h_t, i_t) has a $3\nu_1\rho^{h_t-1}$-optimal parent under the event E_t.

Step 3: Bound on the cumulative regret. We first decompose $\widetilde{R}_n^{\mathcal{E}}$ over different depths. Let $1 \leq \overline{H} \leq H(n)$ a constant

to be chosen later, then we have

$$\begin{aligned}
\widetilde{R}_n^{\mathcal{E}} &= \sum_{t=1}^{n} \Delta_{h_t,i_t} \mathbb{I}_{\mathcal{E}_t} \le \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} \sum_{t=1}^{n} \Delta_{h,i} \mathbb{I}_{(h_t,i_t)=(h,i)} \mathbb{I}_{\mathcal{E}_t} \\
&\overset{(1)}{\le} \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} \sum_{t=1}^{n} 3c\sqrt{\frac{\log(2/\tilde{\delta}(t))}{T_{h,i}(t)}} \mathbb{I}_{(h_t,i_t)=(h,i)} \overset{(2)}{\le} \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} \sum_{s=1}^{T_{h,i}(n)} 3c\sqrt{\frac{\log(2/\tilde{\delta}(\bar{t}_{h,i}))}{s}} \\
&\le \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} \int_1^{T_{h,i}(n)} 3c\sqrt{\frac{\log(2/\tilde{\delta}(\bar{t}_{h,i}))}{s}} ds \le \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} 6c\sqrt{T_{h,i}(n)\log(2/\tilde{\delta}(\bar{t}_{h,i}))} \\
&= \underbrace{6c \sum_{h=0}^{\overline{H}} \sum_{i\in\mathcal{I}_h(n)} \sqrt{T_{h,i}(n)\log(2/\tilde{\delta}(\bar{t}_{h,i}))}}_{(a)} + \underbrace{6c \sum_{h=\overline{H}+1}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} \sqrt{T_{h,i}(n)\log(2/\tilde{\delta}(\bar{t}_{h,i}))}}_{(b)}
\end{aligned} \tag{18}$$

where in (1) we rely on the definition of event E_t and Eq. 16 and in (2) we rely on the fact that at any time step t when the algorithm pulls the arm (h,i), $T_{h,i}$ is incremented by 1 and that by definition of $\bar{t}_{h,i}$ we have that $t \le \bar{t}_{h,i}$. We now bound the two terms in the RHS of Eq. 18. We first simplify the first term as

$$\begin{aligned}
(a) &= \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} \sqrt{T_{h,i}(n)\log(2/\tilde{\delta}(\bar{t}_{h,i}))} \le \sum_{h=0}^{\overline{H}} \sum_{i\in\mathcal{I}_h(n)} \sqrt{\tau_h(n)\log(2/\tilde{\delta}(n))} \\
&= \sum_{h=0}^{\overline{H}} |\mathcal{I}_h(n)| \sqrt{\tau_h(n)\log(2/\tilde{\delta}(n))},
\end{aligned} \tag{19}$$

where the inequality follows from $T_{h,i}(n) \le \tau_h(n)$ and $\bar{t}_{h,i} \le n$. We now need to provide a bound on the number of nodes at each depth h. We first notice that since T is a binary tree, the number of nodes at depth h is at most twice the number of nodes at depth $h-1$ that have been expanded (i.e., the parent nodes), i.e., $|\mathcal{I}_h(n)| \le 2|\mathcal{I}^+_{h-1}(n)|$. We also recall the result of Eq. 17 which guarantees that (h_t^p, i_t^p), the parent of the selected node (h_t, i_t), is $3\nu_1\rho^{h_t-1}$ optimal, that is, HCT never selects a node (h_t, i_t) unless its parent is $3\nu_1\rho^{h_t-1}$ optimal. From Asm. 5 we have that the number of $3\nu_1\rho^h$-optimal nodes is bounded by the covering number $\mathsf{N}(3\nu_1/\nu_2\varepsilon, l, \varepsilon)$ with $\varepsilon = \nu_1\rho^h$. Thus we obtain the bound

$$|\mathcal{I}_h(n)| \le 2|\mathcal{I}^+_{h-1}(n)| \le 2C(\nu_2\rho^{(h-1)})^{-d}, \tag{20}$$

where d is the near-optimality dimension of f around x^*. This bound combined with Eq. 19 implies that

$$\begin{aligned}
(a) &\le \sum_{h=0}^{\overline{H}} 2C\nu_2^{-d}\rho^{-(h-1)d}\sqrt{\tau_h(n)\log(2/\tilde{\delta}(n))} \le \sum_{h=0}^{\overline{H}} 2C\nu_2^{-d}\rho^{-(h-1)d}\sqrt{\frac{c^2\log(1/\tilde{\delta}(n^+))}{\nu_1^2}\rho^{-2h}\log(2/\tilde{\delta}(n))} \\
&\le 2C\nu_2^{-d}\frac{c\log(2/\tilde{\delta}(n^+))}{\nu_1}\rho^d \sum_{h=0}^{\overline{H}} \rho^{-h(d+1)} \le 2C\nu_2^{-d}\frac{c\log(2/\tilde{\delta}(n^+))}{\nu_1}\rho^d \frac{\rho^{-\overline{H}(d+1)}}{1-\rho}.
\end{aligned} \tag{21}$$

We now bound the second term of Eq. 18 as

$$(b) \overset{(1)}{\le} \sqrt{\sum_{h=\overline{H}+1}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} \log(2/\delta(\bar{t}_{h,i}))} \sqrt{\sum_{h=\overline{H}+1}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} T_{h,i}(n)} \overset{(2)}{\le} \sqrt{\sum_{h=\overline{H}+1}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} \log(2/\tilde{\delta}(\bar{t}_{h,i}))} \sqrt{n} \tag{22}$$

where in (1) we make use of Cauchy-Schwarz inequality and in (2) we simply bound the total number of samples by n. We now focus on the summation in the first square root. We recall that we denote by $\tilde{t}_{h,i}$ the last time when any of the two

children of node (h,i) has been pulled. Then we have the following sequence of inequalities.

$$\begin{aligned} n &= \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} T_{h,i}(n) \geq \sum_{h=0}^{H(n)-1} \sum_{i\in\mathcal{I}_h^+(n)} T_{h,i}(n) \geq \sum_{h=0}^{H(n)-1} \sum_{i\in\mathcal{I}_h^+(n)} T_{h,i}(\tilde{t}_{h,i}) \overset{(1)}{\geq} \sum_{h=0}^{H(n)-1} \sum_{i\in\mathcal{I}_h^+(n)} \tau_h(\tilde{t}_{h,i}) \\ &\geq \sum_{h=\overline{H}}^{H(n)-1} \sum_{i\in\mathcal{I}_h^+(n)} \tau_h(\tilde{t}_{h,i}) \geq \sum_{h=\overline{H}}^{H(n)-1} \sum_{i\in\mathcal{I}_h^+(n)} \frac{\rho^{-2h}c^2\log(1/\tilde{\delta}(\tilde{t}^+_{h,i}))}{\nu_1^2} \\ &\geq \frac{c^2\rho^{-2\overline{H}}}{\nu_1^2} \sum_{h=\overline{H}}^{H(n)-1} \rho^{2(\overline{H}-h)} \sum_{i\in\mathcal{I}_h^+(n)} \log(1/\tilde{\delta}(\tilde{t}^+_{h,i})) \overset{(2)}{\geq} \frac{c^2\rho^{-2\overline{H}}}{\nu_1^2} \sum_{h=\overline{H}}^{H(n)-1} \sum_{i\in\mathcal{I}_h^+(n)} \log(1/\tilde{\delta}(\tilde{t}^+_{h,i}))), \end{aligned} \tag{23}$$

where in (1) we rely on the fact that, at each time step t, *HCT-iid* only selects a node when $T_{h,i}(t) \geq \tau_{h,i}(t)$ for its parent and in (2) we used that $\rho^{2(\overline{H}-h)} \geq 1$ for all $h \geq \overline{H}$. We notice that, by definition of $\tilde{t}_{h,i}$, for any internal node (h,i) $\tilde{t}_{h,i} = \max(\bar{t}_{h+1,2i-1}, \bar{t}_{h+1,2i})$. We also notice that for any $t_1, t_2 > 0$ we have that $[\max(t_1,t_2)]^+ = \max(t_1^+, t_2^+)$. This implies that

$$\begin{aligned} n &\geq \frac{c^2\rho^{-2\overline{H}}}{\nu_1^2} \sum_{h=\overline{H}}^{H(n)-1} \sum_{i\in\mathcal{I}_h^+(n)} \log(1/\tilde{\delta}([\max(\bar{t}_{h+1,2i-1}, \bar{t}_{h+1,2i})]^+)) \\ &\overset{(1)}{=} \frac{c^2\rho^{-2\overline{H}}}{\nu_1^2} \sum_{h=\overline{H}}^{H(n)-1} \sum_{i\in\mathcal{I}_h^+(n)} \max(\log(1/\tilde{\delta}(\bar{t}^+_{h+1,2i-1})), \log(1/\tilde{\delta}(\bar{t}^+_{h+1,2i-1}))) \\ &\overset{(2)}{\geq} \frac{c^2\rho^{-2\overline{H}}}{\nu_1^2} \sum_{h=\overline{H}}^{H(n)-1} \sum_{i\in\mathcal{I}_h^+(n)} \frac{\log(1/\tilde{\delta}(\bar{t}^+_{h+1,2i-1})) + \log(1/\tilde{\delta}(\bar{t}^+_{h+1,2i}))}{2} \\ &\overset{(3)}{=} \frac{c^2\rho^{-2\overline{H}}}{2\nu_1^2} \sum_{h'=\overline{H}+1}^{H(n)} \sum_{i\in\mathcal{I}^+_{h'-1}(n)} \log(1/\tilde{\delta}(\bar{t}^+_{h',2i-1})) + \log(1/\tilde{\delta}(\bar{t}^+_{h',2i})) \\ &\overset{(4)}{=} \frac{c^2\rho^{-2\overline{H}}}{2\nu_1^2} \sum_{h'=\overline{H}+1}^{H(n)} \sum_{i'\in\mathcal{I}_{h'}(n)} \log(1/\tilde{\delta}(\bar{t}^+_{h',i'})), \end{aligned} \tag{24}$$

where in (1) we rely on the fact that, for any $t > 0$, $\log(1/\tilde{\delta}(t))$ is an increasing function of t. Therefore we have that $\log(1/\tilde{\delta}(\max(t_1,t_2))) = \max(\log(1/\tilde{\delta}(t_1)), \log(1/\tilde{\delta}(t_2)))$ for any $t_1, t_2 > 0$. In (2) we rely on the fact that the maximum of some random variables is always larger than their average. We introduce a new variable $h' = h+1$ to derive (3). For proving (4) we rely on the argument that, for any $h > 0$, $\mathsf{I}_h^+(n)$ covers all the internal nodes at layer h. This implies that the set of the children of $\mathsf{I}_h^+(n)$ covers $\mathsf{I}_{h+1}(n)$. This combined with fact that the inner sum in (3) is essentially taken on the set of the children of $\mathsf{I}^+_{h'-1}(n)$ proves (4).

Inverting Eq. 24 we have

$$\sum_{h=\overline{H}+1}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} \log(1/\tilde{\delta}(\bar{t}^+_{h,i})) \leq \frac{2\nu_1^2\rho^{2\overline{H}}n}{c^2}. \tag{25}$$

By plugging Eq. 25 into Eq. 22 we deduce

$$\begin{aligned} (b) &\leq \sqrt{\sum_{h=\overline{H}+1}^{H(n)} \sum_{i\in\mathcal{I}_h} \log(2/\tilde{\delta}(\bar{t}^+_{h,i}))}\sqrt{n} \leq \sqrt{\sum_{h=\overline{H}+1}^{H(n)} \sum_{i\in\mathcal{I}_h} 2\log(1/\tilde{\delta}(\bar{t}^+_{h,i}))}\sqrt{n} \\ &\leq \sqrt{\frac{4\nu_1^2\rho^{2\overline{H}}n}{c^2}}\sqrt{n} = \frac{2}{c}\nu_1\rho^{\overline{H}}n. \end{aligned}$$

This combined with Eq. 21 provides the following bound on $\widetilde{R}_n$:

$$\widetilde{R}_n^{\mathcal{E}} \leq 12\nu_1 \left[\frac{Cc^2\nu_2^{-d}\rho^d \log(2/\tilde{\delta}(n))}{\nu_1^2(1-\rho)} \rho^{-\overline{H}(d+1)} + \rho^{\overline{H}} n \right].$$

We then choose $\overline{H}$ to minimize the previous bound. Notably we equalize the two terms in the bound by choosing

$$\rho^{\overline{H}} = \left(\frac{c^2 C \nu_2^{-d}\rho^d}{(1-\rho)\nu_1^2} \frac{\log(2/\tilde{\delta}(n))}{n} \right)^{\frac{1}{d+2}},$$

which, once plugged into the previous regret bound, leads to

$$\widetilde{R}_n^{\mathcal{E}} \leq \frac{24\nu_1}{c} \left(\frac{c^2 C \nu_2^{-d}\rho^d}{(1-\rho)\nu_1^2} \right)^{\frac{1}{d+2}} \left(\log(2/\tilde{\delta}(n)) \right)^{\frac{1}{d+2}} n^{\frac{d+1}{d+2}}.$$

Using the values of $\tilde{\delta}(t)$ and c defined in Lemma 3, the previous expression becomes

$$\widetilde{R}_n^{\mathcal{E}} \leq 3 \left(\frac{2^{2(d+3)}\nu_1^{2(d+1)} C \nu_2^{-d}\rho^d}{(1-\rho)^{d/2+3}} \right)^{\frac{1}{d+2}} \left(\log\left(\frac{2n}{\delta} \sqrt[8]{\frac{3\nu_1}{\rho}} \right) \right)^{\frac{1}{d+2}} n^{\frac{d+1}{d+2}}.$$

This combined with the regret bound of Eq. 9 and the result of Lem. 4 and a union bound on all $n \in \{1, 2, 3, \dots\}$ proves the final result with a probability at least $1 - \delta$.

□

B. Correlated Bandit feedback

We begin the analysis of *HCT*-Γ by proving some useful concentration inequalities for non-iid random variables under the mixing assumptions of Sect. 2.

B.1. Concentration Inequality for non-iid Episodic Random Variables

In this section we extend the result in (Azar et al., 2013) and we derive a concentration inequality for averages of non-iid random variables grouped in episodes. In fact, given the structure of the *HCT*-Γ algorithm, the rewards observed from an arm x are not necessarily consecutive but they are obtained over multiple episodes. This result is of independent interest, thus we first report it in its general form and we later apply it to *HCT*-Γ.

In *HCT*-Γ, once an arm is selected, it is pulled for a number of consecutive steps and many steps may pass before it is selected again. As a result, the rewards observed from one arm are obtained through a series of episodes. Given a fixed horizon n, let $K_n(x)$ be the total number of episodes when arm x has been selected, we denote by $t_k(x)$, with $k = 1, \dots, K_n(x)$, the step when k-th episode of arm x has started and by $v_k(x)$ the length of episode k. Finally, $T_n(x) = \sum_k^{K_n(x)} v_k(x)$ is the total number of samples from arm x. The objective is to study the concentration of the empirical mean built using all the samples

$$\widehat{\mu}_n(x) = \frac{1}{T_n(x)} \sum_{k=1}^{K_n(x)} \sum_{t=t_k(x)}^{t_k(x)+v_k(x)} r_t(x),$$

towards the mean-reward $f(x)$ of the arm. In order to simplify the notation, in the following we drop the dependency from n and x and we use K, t_k, and v_k. We first introduce two quantities. For any $t = 1, \dots, n$ and for any $k = 1, \dots, K$, we define

$$M_t^k(x) = \mathbb{E}\Big[\sum_{t'=t_k}^{t_k+v_k} r_{t'} \big| \mathsf{F}_t \Big],$$

as the expectation of the sum of rewards within episode k, conditioned on the filtration F_t up to time t (see definition in Section 2),[9] and the residual

$$\varepsilon_t^k(x) = M_t^k(x) - M_{t-1}^k(x).$$

We prove the following.

Lemma 5. *For any $x \in \mathsf{X}$, $k = 1, \dots, K$, and $t = 1, \dots, n$, $\varepsilon_t^k(x)$ is a bounded martingale sequence difference, i.e., $\varepsilon_t^k(x) \leq 2\Gamma + 1$ and $\mathbb{E}[\varepsilon_t^k(x) | \mathsf{F}_{t-1}] = 0$.*

Proof. Given the definition of $M_t^k(x)$ we have that

$$\begin{aligned}
\varepsilon_t^k(x) = M_t^k(x) - M_{t-1}^k(x) &= \mathbb{E}\Big[\sum_{t'=t_k}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_t\Big] - \mathbb{E}\Big[\sum_{t'=t_k}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_{t-1}\Big] \\
&= \sum_{t'=t_k}^{t} r_{t'} + \mathbb{E}\Big[\sum_{t'=t+1}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_t\Big] - \sum_{t'=t_k}^{t-1} r_{t'} - \mathbb{E}\Big[\sum_{t'=t}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_{t-1}\Big] \\
&= r_t + \mathbb{E}\Big[\sum_{t'=t+1}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_t\Big] - \mathbb{E}\Big[\sum_{t'=t}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_{t-1}\Big] \\
&= r_t - f(x) + \mathbb{E}\Big[\sum_{t'=t+1}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_t\Big] - (t_k + v_k - t) f(x) + (t_k + v_k - t + 1) f(x) - \mathbb{E}\Big[\sum_{t'=t}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_{t-1}\Big] \\
&\leq 1 + \Gamma + \Gamma.
\end{aligned}$$

Since the previous inequality holds both ways, we obtain that $|\varepsilon_t^k(x)| \leq 2\Gamma + 1$. Furthermore, we have that

$$\begin{aligned}
\mathbb{E}\big[\varepsilon_t^k(x) | \mathsf{F}_{t-1}\big] &= \mathbb{E}\big[M_t^k(x) - M_{t-1}^k(x) | \mathsf{F}_{t-1}\big] \\
&= \mathbb{E}\Big[r_t + \mathbb{E}\Big[\sum_{t'=t+1}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_t\Big] \Big| \mathsf{F}_{t-1}\Big] - \mathbb{E}\Big[\sum_{t'=t}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_{t-1}\Big] = 0.
\end{aligned}$$

□

We can now proceed to derive a high-probability concentration inequality for the average reward of each arm x.

Lemma 6. *For any $x \in \mathsf{X}$ pulled $K(x)$ episodes, each of length $v_k(x)$, for a total number of $T(x)$ samples, we have that*

$$\left| \frac{1}{T(x)} \sum_{k=1}^{K(x)} \sum_{t=t_k}^{t_k+v_k} r_t - f(x) \right| \leq (2\Gamma + 1) \sqrt{\frac{2\log(2/\delta)}{T(x)}} + \frac{K(x)\Gamma}{T(x)}, \tag{26}$$

with probability $1 - \delta$.

Proof. We first notice that for any episode k[10]

$$\sum_{t=t_k}^{t_k+v_k} r_t = M_{t_k+v_k}^k,$$

since $M_{t_k+v_k}^k = \mathbb{E}\Big[\sum_{t'=t_k}^{t_k+v_k} r_{t'} \Big| \mathsf{F}_{t_k+v_k}\Big]$ and the filtration completely determines all the rewards. We can further develop the previous expression using a telescopic expansion which allows us to rewrite the sum of the rewards as a sum of residuals

[9] Notice that the index t of the filtration can be before, within, or after the k-th episode.

[10] We drop the dependency of M on x.

ε_t^k as

$$\sum_{t=t_k}^{t_k+v_k} r_t = M_{t_k+v_k}^k = M_{t_k+v_k}^k - M_{t_k+v_k-1}^k + M_{t_k+v_k-1}^k - M_{t_k+v_k-2}^k + M_{t_k+v_k-2}^k + \cdots - M_{t_k}^k + M_{t_k}^k$$
$$= \varepsilon_{t_k+v_k}^k + \varepsilon_{t_k+v_k-1}^k + \cdots + \varepsilon_{t_k+1}^k + M_{t_k}^k = \sum_{t=t_k+1}^{t_k+v_k} \varepsilon_t^k + M_{t_k}^k.$$

Thus we can proceed by bounding

$$\left| \sum_{k=1}^{K(x)} \Big(\sum_{t=t_k}^{t_k+v_k} r_t - v_k f(x) \Big) \right| \le \left| \sum_{k=1}^{K(x)} \sum_{t=t_k+1}^{t_k+v_k} \varepsilon_t^k \right| + \left| \sum_{k=1}^{K(x)} \Big(M_{t_k}^k - v_k f(x) \Big) \right|$$
$$\le \left| \sum_{k=1}^{K(x)} \sum_{t=t_k+1}^{t_k+v_k} \varepsilon_t^k \right| + K(x)\Gamma.$$

By Lem. 5 ε_t^k is a bounded martingale sequence difference, thus we can directly apply the Azuma's inequality and obtain that

$$\left| \sum_{k=1}^{K(x)} \sum_{t=t_k+1}^{t_k+v_k} \varepsilon_t^k \right| \le (2\Gamma+1)\sqrt{2T(x)\log(2/\delta)}.$$

Grouping all the terms together and dividing by $T(x)$ leads to the statement. □

B.2. Proof of Thm. 2

The notation needed in this section is the same as in Section A. We only need to restate the notation about the episodes from previous section to *HCT*-Γ. We denote by $K_{h,i}(n)$ the number of episodes for node (h,i) up to time n, by $t_{h,i}(k)$ the step when episode k is started, and by $v_{h,i}(k)$ the number of steps of episode k.

We first notice that Lemma 1 holds unchanged also for *HCT*-Γ, thus bounding the maximum depth of an *HCT* tree to $H(n) \le H_{\max}(n) = \frac{1}{1-\rho}\log\left(\frac{n\nu_1^2}{2(c\rho)^2}\right)$. We begin the main analysis by applying the result of Lem. 6 to bound the estimation error of $\widehat{\mu}_{h,i}(t)$ at each time step t.

Lemma 2. *Under assumptions 1 and 2, for any fixed node (h,i) and step t, we have that*

$$|\widehat{\mu}_{h,i}(t) - f(x_{h,i})| \le (3\Gamma+1)\sqrt{2\frac{\log(5/\delta)}{T_{h,i}(t)}} + \frac{\Gamma\log(t)}{T_{h,i}(t)}.$$

with probability $1-\delta$. Furthermore, the previous expression can be conveniently restated for any $0 < \varepsilon \le 1$ as

$$\mathbb{P}(|\widehat{\mu}_{h,i}(t) - f(x_{h,i})| > \epsilon) \le 5t^{1/3}\exp\left(-\frac{T_{h,i}(t)\varepsilon^2}{2(3\Gamma+1)^2}\right).$$

Proof. As a direct consequence of Lem. 6 we have w.p. $1-\delta$,

$$|\widehat{\mu}_{h,i}(t) - f(x_{h,i})| \le (2\Gamma+1)\sqrt{\frac{2\log(2/\delta)}{T_{h,i}(t)}} + \frac{K_{h,i}(t)\Gamma}{T_{h,i}(t)},$$

where $K_{h,i}(t)$ is the number of episodes in which we pull arm $x_{h,i}$. At each episode in which $x_{h,i}$ is selected, its number of pulls $T_{h,i}$ is doubled w.r.t. the previous episode, except for those episodes where the current time s becomes larger than s^+, which triggers the termination of the episode. However since s^+ doubles whenever s becomes larger than s^+, the total number of times when episodes are interrupted because of $s \ge s^+$ can be at maximum $\log_2(t)$ withing a time horizon of

t. This means that the total number of times an episode finishes without doubling $T_{h,i}(t)$ is bounded by $\log_2(t)$. Thus we have

$$T_{h,i}(t) \geq \sum_{k=1}^{K_{h,i}(t)-\log_2(t)-1} 2^{k-1} \geq 2^{K_{h,i}(t)-\log_2(t)-2},$$

where in the second inequality we simply keep the last term of the summation. Inverting the previous inequality we obtain that

$$K_{h,i}(t) \leq \log_2(4T_{h,i}(t)) + \log_2(t),$$

which bounds the number of episodes w.r.t. the number of pulls and the time horizon t. Combining this result with the high probability bound of Lem. 6, we obtain

$$|\widehat{\mu}_{h,i}(t) - f(x_{h,i})| \leq (2\Gamma+1)\sqrt{\frac{2\log(2/\delta)}{T_{h,i}(t)}} + \Gamma\frac{\log_2(4T_{h,i}(t))}{T_{h,i}(t)} + \Gamma\frac{\log(t)}{T_{h,i}(t)},$$

with probability $1-\delta$. The statement of the Lemma is obtained by further simplifying the second term in the right hand side with the objective of achieving a more homogeneous expression. In particular, we have that

$$\log_2(4T_{h,i}(t)) = 2\log_2(2\sqrt{T_{h,i}(t)}) = 2(\log_2(\sqrt{T_{h,i}(t)})+1) \leq 2\sqrt{T_{h,i}(t)},$$

and

$$\begin{aligned}|\widehat{\mu}_{h,i}(t) - f(x_{h,i})| &\leq (2\Gamma+1)\sqrt{\frac{2\log(2/\delta)}{T_{h,i}(t)}} + \frac{2\Gamma\sqrt{T_{h,i}(t)}}{T_{h,i}(t)} + \frac{\Gamma\log(t)}{T_{h,i}(t)} \\ &\leq (3\Gamma+1)\sqrt{\frac{2\log(5/\delta)}{T_{h,i}(t)}} + \frac{\Gamma\log(t)}{T_{h,i}(t)}.\end{aligned}$$

To prove the second statement we choose $\varepsilon := (3\Gamma+1)\sqrt{\frac{2\log(5/\delta)}{T_{h,i}(t)}} + \frac{\Gamma\log(t)}{T_{h,i}(t)}$ and we solve the previous expression w.r.t. δ:

$$\delta = 5\exp\left[-\frac{T_{h,i}(t)(\varepsilon - \Gamma\log(t)/T_{h,i}(t))^2}{2(3\Gamma+1)^2}\right].$$

The following sequence of inequalities then follows

$$\begin{aligned}\mathbb{P}(|\widehat{\mu}_{h,i}(t) - f(x_{h,i})| > \varepsilon) \leq \delta = 5\exp\left[-\frac{T_{h,i}(t)(\varepsilon - \Gamma\log(t)/T_{h,i}(t))^2}{2(3\Gamma+1)^2}\right] &\leq 5\exp\left[-\frac{T_{h,i}(t)(\varepsilon^2 - 2\varepsilon\Gamma\log(t)/T_{h,i}(t))}{2(3\Gamma+1)^2}\right] \\ &\leq 5\exp\left[-\frac{T_{h,i}(t)(\varepsilon^2 - 2\Gamma\log(t)/T_{h,i}(t))}{2(3\Gamma+1)^2}\right] = 5\exp\left[-\frac{T_{h,i}(t)\varepsilon^2}{(3\Gamma+1)^2} + \frac{2\Gamma\log(t)}{2(3\Gamma+1)^2}\right] \\ &\leq 5\exp\left[-\frac{T_{h,i}(t)\varepsilon^2}{(3\Gamma+1)^2} + \frac{2\Gamma\log(t)}{12\Gamma}\right] = 5\exp\left[-\frac{T_{h,i}(t)\varepsilon^2}{2(3\Gamma+1)^2} + \log(t^{1/6})\right],\end{aligned}$$

which concludes the proof. □

The result of Lem. **??** facilitates the adaption of the previous results of iid case to the case of correlated rewards, since this bound is similar to those of standard tail's inequality such as Hoeffding and Azuma's inequality. Based on this result we can extend the results of previous section to the case of dependent arms.

We now introduce the high probability event $\mathsf{E}_{t,n}$ under which the mean reward for all the selected nodes in the interval $[t,n]$ is within a confidence interval of the empirical estimates at every time step in the interval. The event $\mathsf{E}_{t,n}$ is needed to concentrate the sum of obtained rewards around the sum of their corresponding arm means. Note that unlike the previous theorem where we could make use of a simple martingale argument to concentrate the rewards around their means, here the rewards are not unbiased samples of the arm means. Therefore, we need a more advanced technique than the Azuma's inequality for concentration of measure.

Lemma 7 (High-probability event)**.** *We define the set of all the possible nodes in trees of maximum depth $H_{\max}(t)$ as*

$$\mathsf{L}_t = \bigcup_{\mathcal{T}:\mathit{Depth}(\mathcal{T})\leq H_{\max}(t)} \mathit{Nodes}(\mathsf{T}).$$

We introduce the event

$$\Omega_t = \left\{\forall (h,i) \in \mathsf{L}_t, \forall T_{h,i}(t) = 1,\dots,t : \left|\widehat{\mu}_{h,i}(t) - f(x_{h,i})\right| \leq c\sqrt{\frac{\log(1/\tilde{\delta}(t))}{T_{h,i}(t)}}\right\},$$

where $x_{h,i} = \mathsf{P}_{h,i}$ is the arm corresponding to node (h,i), and the event $\mathsf{E}_{t,n} = \bigcap_{s=t}^{n} \Omega_s$. If

$$c = 6(3\Gamma+1)\sqrt{\frac{1}{1-\rho}} \quad \textit{and} \quad \tilde{\delta}(t) = \frac{\delta}{t}\sqrt[9]{\frac{\rho}{4\nu_1}},$$

then for any fixed t, the event Ω_t holds with probability $1-\delta/t^7$ and the joint event $\mathsf{E}_{t,n}$ holds with probability at least $1-\delta/(6t^6)$.

Proof. We upper bound the probability of complementary event of Ω_t after t steps

$$\begin{aligned}
\mathbb{P}[\Omega_t^{\mathsf{c}}] &= \sum_{(h,i)\in\mathcal{L}_t}\sum_{T_{h,i}(t)=1}^{t} \mathbb{P}\left[\left|\widehat{\mu}_{h,i}(t) - f(x_{h,i})\right| \geq c\sqrt{\frac{\log(1/\tilde{\delta}(t))}{T_{h,i}(t)}}\right] \\
&\leq \sum_{(h,i)\in\mathcal{L}_t}\sum_{T_{h,i}(t)=1}^{t} 5t^{1/3}\exp\left(-T_{h,i}(t)c^2\frac{\log(1/\tilde{\delta}(t))}{(3\Gamma+1)^2 T_{h,i}(t)}\right) \\
&\leq 5\exp(-c^2/(3\Gamma+1)^2\log(1/\tilde{\delta}(t)))t^{4/3}|\mathsf{L}_t|,
\end{aligned}$$

Similar to the proof of Lem. 4, we have that $|\mathsf{L}_t| \leq 2^{H_{\max}(t)+1}$. Thus

$$\mathbb{P}[\Omega_t^{\mathsf{c}}] \leq 5(\tilde{\delta}(t))^{(c/(3\Gamma+1))^2} t^{4/3} 2^{H_{\max}(t)+1}.$$

We first derive a bound on the the term $2^{H_{\max}(t)}$ as

$$2^{H_{\max}(t)} \leq \mathrm{pow}\left(2, \log_2\left(\frac{t\nu_1^2}{2(c\rho)^2}\right)^{\frac{1}{2\log_2(e)(1-\rho)}}\right) \leq \left(\frac{t\nu_1^2}{2(c\rho)^2}\right)^{\frac{1}{2(1-\rho)}},$$

where we used the definition of the upper bound $H_{\max}(t)$. which leads to

$$\mathbb{P}[\Omega_t^{\mathsf{c}}] \leq 10t^{4/3}\big(\tilde{\delta}(t)\big)^{(c/(3\Gamma+1))^2}\left(\frac{t\nu_1^2}{2(c\rho)^2}\right)^{\frac{1}{2(1-\rho)}}.$$

The choice of c and $\tilde{\delta}(t)$ as in the statement leads to $\mathbb{P}[\Omega_t^{\mathsf{c}}] \leq \frac{\delta}{t^7}$ (steps are similar to Lemma 3) .

The bound on the joint event $\mathsf{E}_{t,n}$ follows from a union bound as

$$\mathbb{P}\left[\mathsf{E}_{t,n}^{\mathsf{c}}\right] = \mathbb{P}\Big[\bigcup_{s=t}^{n}\Omega_s^{\mathsf{c}}\Big] \leq \sum_{s=t}^{n}\mathbb{P}(\Omega_s^{\mathsf{c}}) \leq \int_t^{\infty}\frac{\delta}{s^7}ds = \frac{\delta}{6t^6}.$$

□

Recalling the definition of regret from Sect. 2, we decompose the regret of *HCT*-iid in two terms depending on whether event E_t holds or not (i.e., failing confidence intervals). Let the instantaneous regret be $\Delta_t = f^* - r_t$, then we rewrite the regret as

$$R_n = \sum_{t=1}^{n}\Delta_t = \sum_{t=1}^{n}\Delta_t\mathbb{I}_{\mathcal{E}_t} + \sum_{t=1}^{n}\Delta_t\mathbb{I}_{\mathcal{E}_t^{\mathsf{c}}} = R_n^{\mathcal{E}} + R_n^{\mathcal{E}^{\mathsf{c}}}. \tag{27}$$

We first study the regret in the case of failing confidence intervals.

Lemma 8 (Failing confidence intervals)**.** *Given the parameters c and $\tilde{\delta}(t)$ as in Lemma 7, the regret of HCT-iid when confidence intervals fail to hold is bounded as*

$$R_n^{\mathcal{E}^c} \leq \sqrt{n},$$

with probability $1 - \frac{\delta}{30n^2}$.

Proof. The proof is the same as in Lemma 4 expect for the union bound which is applied to $\mathsf{E}_{t,n}$ for $t = \sqrt{n}, \dots, n$. □

We are now ready to prove the main theorem, which only requires to study the regret term under events $\{\mathsf{E}_{t,n}\}$.

Theorem 2 (Regret bound of *HCT*-Γ). *Let $\delta \in (0,1)$, $\tilde{\delta}(t) = \sqrt[3]{\rho/(3\nu_1)}\delta/t$, and $c = 6(3\Gamma+1)\sqrt{1/(1-\rho)}$. We assume that assumptions 1–5 hold and that rewards are generated according to the general model defined in Section 2. Then the regret of HCT-iid after n steps is*

$$R_n \leq 3\left(\frac{2^{2d+7}\nu_1^{2(d+1)}C\nu_2^{-d}\rho^d}{(1-\rho)^{d+7}}\right)^{\frac{1}{d+2}}\left(\log\left(\frac{2n}{\delta}\sqrt[8]{\frac{3\nu_1}{\rho}}\right)\right)^{\frac{1}{d+2}} n^{\frac{d+1}{d+2}} + 2\sqrt{n\log(4n/\delta)},$$

with probability $1-\delta$.

Proof. The structure of the proof is exactly the same as in Theorem 1. Thus, here we report only the main differences in each step.

Step 1: Decomposition of the regret. We first decompose the regret in two terms. We rewrite the instantaneous regret Δ_t as

$$\Delta_t = f^* - r_t = f^* - f(x_{h_t,i_t}) + f(x_{h_t,i_t}) - r_t = \Delta_{h_t,i_t} + \widehat{\Delta}_t,$$

which leads to a regret

$$R_n^{\mathcal{E}} = \sum_{t=1}^n \Delta_{h_t,i_t}\mathbb{I}_{\mathcal{E}_{t,n}} + \sum_{t=1}^n \widehat{\Delta}_t\mathbb{I}_{\mathcal{E}_{t,n}} = \widetilde{R}_n^{\mathcal{E}} + \widehat{R}_n^{\mathcal{E}}. \tag{28}$$

Unlike in Theorem 1, the definition of $\widehat{R}_n^{\mathcal{E}}$ still requires the event $\mathbb{I}_{\mathcal{E}_{t,n}}$ and the sequence $\{\widehat{\Delta}_t\}_{t=1}^n$ is no longer a bounded martingale difference sequence. In fact, $\mathbb{E}(\widehat{\Delta}_t|\mathsf{F}_{t-1}) \neq 0$ since the expected value of r_t does not coincide with the mean-reward value of the corresponding node $f(x_{h_t,i_t})$. This prevents from directly using the Azuma inequality and extra care is needed to derive a bound. We have that

$$\begin{aligned}
\widehat{R}_n^{\mathcal{E}} &= \sum_{t=1}^n \widehat{\Delta}_t\mathbb{I}_{\mathcal{E}_{t,n}} \leq \sum_{h=0}^{H(n)}\sum_{i\in\mathcal{I}_h(n)}\sum_{t=1}^n \widehat{\Delta}_t\mathbb{I}_{\mathcal{E}_{t,n}}\mathbb{I}_{(h_t,i_t)=(h,i)} \\
&= \sum_{h=0}^{H(n)}\sum_{i\in\mathcal{I}_h(n)}\sum_{t=1}^n (f(x_{h,i}) - r_t)\mathbb{I}_{\mathcal{E}_{t,n}}\mathbb{I}_{(h_t,i_t)=(h,i)} \overset{(1)}{\leq} \sum_{h=0}^{H(n)}\sum_{i\in\mathcal{I}_h(n)}\sum_{t=1}^n (f(x_{h,i}) - r_t)\mathbb{I}_{\Omega_{t_{h,i},n}}\mathbb{I}_{(h_t,i_t)=(h,i)} \\
&\overset{(2)}{=} \sum_{h=0}^{H(n)}\sum_{i\in\mathcal{I}_h(n)} T_{h,i}(\bar{t}_{h,i})(f(x_{h,i}) - \widehat{\mu}_{h,i}(\bar{t}_{h,i}))\mathbb{I}_{\Omega_{\bar{t}_{h,i}}} \\
&\overset{(3)}{\leq} \sum_{h=0}^{H(n)}\sum_{i\in\mathcal{I}_h(n)} cT_{h,i}(\bar{t}_{h,i})\sqrt{\frac{\log(2/\tilde{\delta}(\bar{t}_{h,i}))}{T_{h,i}(\bar{t}_{h,i})}} \leq \sum_{h=0}^{H(n)}\sum_{i\in\mathcal{I}_h(n)} c\sqrt{T_{h,i}(\bar{t}_{h,i})\log(2/\tilde{\delta}(\bar{t}_{h,i}))} \\
&\leq \underbrace{c\sum_{h=0}^{\overline{H}}\sum_{i\in\mathcal{I}_h(n)}\sqrt{T_{h,i}(n)\log(2/\tilde{\delta}(\bar{t}_{h,i}))}}_{(a)} + \underbrace{c\sum_{h=\overline{H}+1}^{H(n)}\sum_{i\in\mathcal{I}_h(n)}\sqrt{T_{h,i}(n)\log(2/\tilde{\delta}(\bar{t}_{h,i}))}}_{(b)},
\end{aligned} \tag{29}$$

where (1) follows from the definition of $\mathsf{E}_{t,n} = \bigcap_{s=t}^{n} \Omega_s$, thus if $\mathsf{E}_{t,n}$ holds at time t then Ω_s also holds at $s = \bar{t}_{h,i} \geq t$. Step (2) follows from the definition of $\widehat{\mu}_{h,i}$: First we notice that for the node (h_n, i_n) we have that $T_{h_n,i_n}(n)\widehat{\mu}_{h_n,i_n}(n) = \sum_{t=1}^{n} r_t \mathbb{I}_{(h_t,i_t)=(h_n,i_n)}$ since we update the statistics at the end. for every other node we have that the last selection time $\bar{t}_{h,i}$ and the end of last episode coincides together . Now since we update the statistics of the selected node at the end of every episode, thus, we have that $T_{h,i}(\bar{t}_{h,i})\widehat{\mu}_{h,i}(\bar{t}_{h,i}) = \sum_{t=1}^{n} r_t \mathbb{I}_{(h_t,i_t)=(h,i)}$ also for $(h,i) \neq (h_n, i_n)$. Step (3) follows from the definition of Ω_s. The resulting bound matches the one in Eq. 18 up to constants and it can be bound similarly.

$$\widehat{R}_n^{\mathcal{E}} \leq 2\nu_1 \left[\frac{Cc^2 \nu_2^{-d} \rho^d \log(2/\tilde{\delta}(n))}{\nu_1^2 (1-\rho)} \rho^{-\overline{H}(d+1)} + \rho^{\overline{H}} n \right].$$

Step 2: Preliminary bound on the regret of selected nodes. The second step follows exactly the same steps as in the proof of Theorem 1 with the only difference that here we use the high-probability event $\mathsf{E}_{t,n}$. As a result the following inequalities hold for the node (h_t, i_t) selected at time t and its parent (h_t^p, i_t^p)

$$\begin{aligned} \Delta_{h_t,i_t} &\leq 3c \sqrt{\frac{\log(2/\tilde{\delta}(t))}{T_{h_t,i_t}(t)}}. \\ \Delta_{h_t^p,i_t^p} &\leq 3\nu_1 \rho^{h_t - 1}. \end{aligned} \tag{30}$$

Step 3: Bound on the cumulative regret. Unlike in the proof of Theorem 1, the total regret $\widetilde{R}_n^{\mathcal{E}}$ should be analyzed with extra care since here we do not update the selected arm as well as the statistics $T_{h,i}(t)$ and $\widehat{\mu}_{h,i}(t)$ for the the entire length of episode, whereas in Theorem 1 we update at every step. Thus the development of $\widetilde{R}_n^{\mathcal{E}}$ slightly differs from Eq. 18. Let $1 \leq \overline{H} \leq H(n)$ a constant to be chosen later, then we have

$$\begin{aligned} \widetilde{R}_n^{\mathcal{E}} &\overset{(1)}{=} \sum_{t=1}^{n} \Delta_{h_t,i_t} \mathbb{I}_{\mathcal{E}_{t,n}} = \sum_{h=0}^{H(n)} \sum_{i \in \mathcal{I}_h(n)} \sum_{t=1}^{n} \Delta_{h,i} \mathbb{I}_{(h_t,i_t)=(h,i)} \mathbb{I}_{\mathcal{E}_{t,n}} = \sum_{h=0}^{H(n)} \sum_{i \in \mathcal{I}_h(n)} \sum_{k=1}^{K_{h,i}(n)} \sum_{t=t_{h,i}(k)}^{t_{h,i}(k)+v_{h,i}(k)} \Delta_{h,i} \mathbb{I}_{\mathcal{E}_{t,n}} \\ &\overset{(2)}{\leq} \sum_{h=0}^{H(n)} \sum_{i \in \mathcal{I}_h(n)} \sum_{k=1}^{K_{h,i}(n)} \sum_{t=t_{h,i}(k)}^{t_{h,i}(k)+v_{h,i}(k)} \left[3c \sqrt{\frac{\log(2/\tilde{\delta}(t))}{T_{h,i}(t)}} \right] \overset{(3)}{=} \sum_{h=0}^{H(n)} \sum_{i \in \mathcal{I}_h(n)} \sum_{k=1}^{K_{h,i}(n)} v_{h,i}(k) \left[3c \sqrt{\frac{\log(2/\tilde{\delta}(t_{h,i}(k)))}{T_{h,i}(t_{h,i}(k))}} \right] \\ &\leq \sum_{h=0}^{H(n)} \sum_{i \in \mathcal{I}_h(n)} 3c \sqrt{\log(2/\tilde{\delta}(\bar{t}_{h,i}))} \sum_{k=1}^{K_{h,i}(n)} \frac{v_{h,i}(k)}{\sqrt{T_{h,i}(t_{h,i}(k))}} \\ &\overset{(4)}{\leq} 3(\sqrt{2}+1)c \sum_{h=0}^{H(n)} \sum_{i \in \mathcal{I}_h(n)} \sqrt{\log(2/\tilde{\delta}(\bar{t}_{h,i})) T_{h,i}(t_{h,i}(K_{h,i}(n)))} \leq 3(\sqrt{2}+1)c \sum_{h=0}^{H(n)} \sum_{i \in \mathcal{I}_h(n)} \sqrt{\log(2/\tilde{\delta}(\bar{t}_{h,i})) T_{h,i}(n)} \\ &= 3(\sqrt{2}+1)c \underbrace{\sum_{h=0}^{\overline{H}} \sum_{i \in \mathcal{I}_h(n)} \sqrt{T_{h,i}(n) \log(2/\tilde{\delta}(\bar{t}_{h,i}))}}_{(a)} + 3(\sqrt{2}+1)c \underbrace{\sum_{h=\overline{H}+1}^{H(n)} \sum_{i \in \mathcal{I}_h(n)} \sqrt{T_{h,i}(n) \log(2/\tilde{\delta}(\bar{t}_{h,i}))}}_{(b)}, \end{aligned} \tag{31}$$

where the first sequence of equalities in (1) simply follows from the definition of episodes. In (2) we bound the instantaneous regret by Eq. 30. Step (3) follows from the fact that when (h,i) is selected, its statistics, including $T_{h,i}$, are not changed until the end of the episode. Step (4) is an immediate application of Lemma 19 in (Jaksch et al., 2010).

Constants apart the terms (a) and (b) coincides with the terms defined in Eq. 18 and similar bounds can be derived.

Putting the bounds on $\widehat{R}_n^{\mathcal{E}}$ and $\widetilde{R}_n^{\mathcal{E}}$ together leads to

$$R_n^{\mathcal{E}} \leq 2(3\sqrt{(2)}+4)\nu_1 \left[\frac{Cc^2 \nu_2^{-d} \rho^d \log(2/\tilde{\delta}(n))}{\nu_1^2 (1-\rho)} \rho^{-\overline{H}(d+1)} + \rho^{\overline{H}} n \right].$$

It is not difficult to prove that for a suitable choice $\overline{H}$, we obtain the final bound of $O(\log(n)^{1/(d+2)} n^{(d+1)/(d+2)})$ on R_n. This combined with the result of Lem. 7 and a union bound on all $n \in \{1, 2, 3, \dots\}$ proves the final result.

□

B.3. Proof of Thm. 3

Theorem 3 *Let $\delta \in (0,1)$, $\tilde{\delta}(n) = \sqrt[9]{\rho/(4\nu_1)}\delta/n$, and $c = 3(3\Gamma+1)\sqrt{1/(1-\rho)}$. We assume that assumptions 1–5 hold and that rewards are generated according to the general model defined in Section 2. Then if $\delta = 1/n$ the space complexity of HCT-Γ is*

$$\mathbb{E}(\mathsf{N}_n) = O(\log(n)^{2/(d+2)} n^{d/(d+2)}).$$

Proof. We assume that the space requirement for each node (i.e., storing variables such as $\widehat{\mu}_{h,i}$, $T_{h,i}$) is a unit. Let B_t denote the event corresponding to the branching/expansion of the node (h_t, i_t) selected at time t, then the space complexity is $\mathsf{N}_n = \sum_{t=1}^{n} \mathbb{I}_{B_t}$. Similar to the regret analysis, we decompose N_n depending on events $\mathsf{E}_{t,n}$, that is

$$\mathsf{N}_n = \sum_{t=1}^{n} \mathbb{I}_{\mathcal{B}_t}\mathbb{I}_{\mathcal{E}_{t,n}} + \sum_{t=1}^{n} \mathbb{I}_{\mathcal{B}_t}\mathbb{I}_{\mathcal{E}^c_{t,n}} = \mathsf{N}_n^{\mathcal{E}} + \mathsf{N}_n^{\mathcal{E}^c}. \tag{32}$$

Since we are targeting the expected space complexity, we take the expectation of the previous expression and the second term can be easily bounded as

$$\mathbb{E}\big[\mathsf{N}_n^{\mathcal{E}^c}\big] = \sum_{t=1}^{n} \mathbb{I}_{\mathcal{B}_t}\mathbb{P}[\mathsf{E}^c_{t,n}] \le \sum_{t=1}^{n} \mathbb{P}[\mathsf{E}^c_t] \le \sum_{t=1}^{n} \frac{\delta}{6t^6} \le C, \tag{33}$$

where the last inequality follows from Lemma 7 and C is a constant independent from n. We now focus on the first term $\mathsf{N}_n^{\mathcal{E}}$. We first rewrite it as the total number of nodes $|\mathsf{T}_n|$ generated by *HCT* over n steps. For any depth $\overline{H} > 0$ we have

$$\mathsf{N}_n^{\mathcal{E}} = \sum_{h=0}^{H(n)} |\mathsf{I}_h(n)| = 1 + \sum_{h=1}^{\overline{H}} |\mathsf{I}_h(n)| + \sum_{h=\overline{H}+1}^{H(n)} |\mathsf{I}_h(n)| \le 1 + \underbrace{\overline{H}|\mathsf{I}_{\overline{H}}(n)|}_{(c)} + \underbrace{\sum_{h=\overline{H}+1}^{H(n)} |\mathsf{I}_h(n)|}_{(d)}. \tag{34}$$

A bound on term (d) can be recovered through the following sequence of inequalities

$$\begin{aligned} n = \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h(n)} T_{h,i}(n) &\ge \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h^+(n)} T_{h,i}(n) \overset{(1)}{\ge} \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h^+(n)} \tau_{h,i}(t_{h,i}) \\ &\overset{(2)}{\ge} \sum_{h=0}^{H(n)} \sum_{i\in\mathcal{I}_h^+(n)} \frac{c^2}{\nu_1^2}\rho^{-2h} \overset{(3)}{\ge} \frac{1}{\nu_1^2} \sum_{h=\overline{H}}^{H(n)-1} |\mathsf{I}_h^+(n)|\rho^{-2h} = \frac{1}{\nu_1^2}\rho^{-2\overline{H}} \sum_{h=\overline{H}}^{H(n)-1} |\mathsf{I}_h^+(n)|\rho^{2(\overline{H}-h)} \\ &\ge \frac{1}{\nu_1^2}\rho^{-2\overline{H}} \sum_{h=\overline{H}}^{H(n)-1} |\mathsf{I}_h^+(n)| \overset{(4)}{\ge} \frac{1}{2\nu_1^2}\rho^{-2\overline{H}} \sum_{h=\overline{H}+1}^{H(n)} |\mathsf{I}_h(n)|, \end{aligned} \tag{35}$$

where (1) follows from the fact that nodes in $\mathsf{I}_h^+(n)$ have been expanded at time $t_{h,i}$ when their number of pulls $T_{h,i}(t_{h,i}) \le T_{h,i}(n)$ exceeded the threshold $\tau_{h,i}(t_{h,i})$. Step (2) follows from Eq. 6, while (3) from the definition of $c > 1$. Finally, step (4) follows from the fact that the number of nodes at depth h cannot be larger than twice the parent nodes at depth $h-1$. By inverting the previous inequality, we obtain

$$(d) \le 2\nu_1^2 n\rho^{2\overline{H}}.$$

On other hand, in order to bound (c), we need to use the same the high-probability events $\mathsf{E}_{t,n}$ and similar passages as in Eq. 20, which leads to $|\mathsf{I}_h(n)| \le 2|\mathsf{I}_{h-1}^+(n)| \le 2C(\nu_2\rho^{(h-1)})^{-d}$. Plugging these results back in Eq. 34 leads to

$$\mathsf{N}_n^{\mathcal{E}} \le 1 + 2\overline{H}C(\nu_2\rho^{(\overline{H}-1)})^{-d} + 2\nu_1^2 n\rho^{2\overline{H}},$$

with high probability. Together with $\mathsf{N}_n^{\mathcal{E}^c}$ we obtain

$$\mathbb{E}\big[\mathsf{N}_n\big] \le 1 + 2\overline{H}C(\nu_2\rho^{(\overline{H}-1)})^{-d} + 2\nu_1^2 n\rho^{2\overline{H}} + C \le 1 + 2H_{\max}(n)C(\nu_2\rho^{(\overline{H}-1)})^{-d} + 2\nu_1^2 n\rho^{2\overline{H}} + C,$$

where $H_{\max}(n)$ is the upper bound on the depth of the tree in Lemma 1. Optimizing $\overline{H}$ in the remaining terms leads to the statement. □

C. Numerical Results

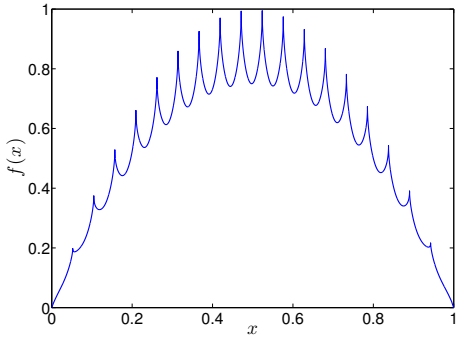

Figure 3. The garland function.

While our primary contribution is the technical analysis just presented, we also give some preliminary simulation results to demonstrate some of HCT's properties.

For our first experiment, we focus on minimizing the regret across repeated function evaluations of the garland function $f(x) = x(1-x)(4-\sqrt{|\sin(60x)|})$ (see Figure 3 in the supplementary material) relative to repeatedly selecting its global optima x^*. Pulling an arm x produces a reward of $f(x) + \varepsilon$, where ε is drawn randomly from the interval $[0, 1]$. These rewards are independent and identically distributed given the selected arm x. We select this function due to its several interesting properties. First, it contains many local optima. Second, around its global optima x^*, it is locally smooth: in particular it behaves as $f^* - c|x - x^*|^\alpha$, for $c = 2$ and $\alpha = 1/2$. And third, it is also possible to show that the near-optimality dimension d of f equals 0.

In this first example we compare *HCT*-iid to the truncated hierarchical optimistic optimization (T-HOO) algorithm (Bubeck et al., 2011a). T-HOO is a state-of-the-art approach for stochastic online optimization, and was developed as a computationally-efficient approach for optimizing a nonlinear function with iid-noisy observations. We evaluate the performances of each algorithm in terms of the per-step regret, $\widetilde{R}_n = R_n/n$. Each run is $n = 10^5$ steps and we average the performance on 10 runs. For both HCT and T-HOO we introduce a tuning parameter used to multiply the upper bounds, and vary this constant per algorithm to maximize the empirical reward.

In Figure 6 we show the per-step regret, the runtime, and the space requirements of each approach. As predicted by the theoretical bounds, the per-step regret $\widetilde{R}_n$ of both *HCT*-iid and truncated *HOO* decrease rapidly with number of steps. Though the big O theoretical bounds are identical for both approaches, empirically we observe in this example that *HCT*-iid outperforms *T-HOO* by a large margin. Similarly, though the computational complexity of both approaches matches in the dependence on the number of time steps, empirically we observe that our approach outperforms *T-HOO* (Figure **??**). Perhaps the most significant expected advantage of HCT-iid over T-HOO for iid settings is in the space requirements. HCT-iid has a space requirement for this domain that scales logarithmically with the time step n, as predicted by Theorem 3 (since the near-optimality dimension $d = 0$). In contrast, a brief analysis of *T-HOO* suggests that its space requirements can grow polynomially, and indeed in this domain we observe such a polynomial grow in memory usage. These patterns mean that *HCT*-iid can achieve a very small regret using a decision tree which contains only few hundred nodes, whereas truncated *HOO* requires to build a much larger tree with orders of magnitude more nodes than *HCT*-iid.

We next consider a simulation for the correlated setting. To do so we create a continuous-state-action Markov decision problem out of the previously described Garland function. There is now a current state of the environment s. Upon taking continuous-valued action x, the state of the environment changes deterministically to $s_{t+1} = (1-\beta)s_t + \beta x$, where we set $\beta = 0.2$. The agent receives a stochastic reward for being in state s, which is (the Garland function) $f(s) + \varepsilon$, where as before ε is drawn randomly fro $[0, 1]$. The initial state s_0 is also drawn randomly from $[0, 1]$. A priori, the agent does not know the transition or reward function, making this a reinforcement learning problem. Though not a standard benchmark RL instance, this problem has multiple local optima and therefore is a interesting case for policy search. In this setting we again our *HCT*-Γ algorithm to a PoWER, a standard powerful RL policy search algorithm (Kober & Peters, 2011). PoWER uses an Expectation Maximization approach to optimize the policy parameters and is therefore not guaranteed to find the global optima. We also compare our algorithm with T-HOO, though this algorithm is specifically designed for iid

setting and one may expect that it may fail to converge to global optima under correlated bandit feedback. As in the iid domain, we include tuning parameters for the upper bounds of the stochastic optimization approaches, and for the window for computing the weighted average in the PoWER method, and optimize over these parameters to maximize performance.

Figure 6 shows per-step regret of the 3 approaches in the MDP. Only *HCT*-Γ succeeds in finding the globally optimal policy, as is evident because only in the case of *HCT*-Γ does the average regret tends to converge to zero (which is as predicted from Theorem 2). The PoWER method finds worse solutions than both stochastic optimization approaches for the same amount of computational time, likely due to using EM which is known to be susceptible to local optima. It's primary advantage is that it has a very small memory requirement. Overall this suggests the benefit of our proposed approach to be used for online MDP policy search, since it quickly (as a function of samples and runtime) can find a global optima, and is, to our knowledge, one of the only policy search methods guaranteed to do so.

D. Application of *HCT* to Policy Search in Markov Decision Problems

As we discussed in Sect. 1, *HCT* may be used for optimization in problems where there exists a strong correlation among the rewards, arm pulls and contexts, at different time steps. An important problem for which *HCT* may be used, is the problem of policy search in infinite-horizon Markov decision processes.A Markov decision process (MDP) M is defined as a tuple $(\mathcal{S}, \mathcal{A}, P)$where $\mathcal{S}$ is the set of states, $\mathcal{A}$ is the set of actions, $P : \mathcal{S} \times \mathcal{A} \to \mathcal{M}(\mathcal{S} \times [0,1])$ is the transition kernel mapping each state-action pair to a distribution over states and rewards. A policy $\pi : \mathcal{S} \to \mathcal{A}$ is a mapping from states to actions. Policy search algorithms (Scherrer & Geist, 2013; Azar et al., 2013; Kober & Peters, 2011) aim at finding the best policy in a policy set with the goal of optimizing some long-term performance measure such as the time average of rewards. Formally, a policy search algorithm operates on the kernel class $\mathcal{G}$corresponding to the class of probability kernels mapping the state space $\mathcal{S}$ to the space of probability measures on $\mathcal{A}$.These methods often assume that every $g \in \mathcal{G}$can be represented by a set of parameters $\theta \in \Theta$, where Θ is a measurable set. Formally, this assumption corresponds to the fact that there exists a policy kernel $\pi_\theta \in \mathcal{G}$mapping the space of states $\mathcal{S}$ to the set of actions $\mathcal{A}$ for any given $\theta \in \Theta$ and vice versa. The learner selects the action $u \in \mathcal{A}$ according to the probability distribution $\pi_\theta(\cdot|s)$ given its current state $s \in \mathcal{S}$ and the policy parameter $\theta \in \Theta$. Any policy $\pi_\theta \in \mathcal{G}$induces a state-reward transition kernel $T : \mathcal{M}(\mathcal{X}) \times \Theta \to \mathcal{M}(\mathcal{X} \times [0,1])$. T relates to the state-reward-action transition kernel P and the policy kernel π_θ as follows

$$T(s', r|s, \theta) := \int_{u \in \mathcal{A}} P(s', r|s, u)\pi_\theta(u|s)du,$$

for all $s, s' \in \mathcal{S}$, $r \in [0,1]$ and $\theta \in \Theta$. For any $\pi_\theta \in \mathcal{G}$and the initial state $s_0 \in S$, the time-average reward $\mu_\theta^\pi(s_0, n)$ obtained over n steps for a given parameter θ is defined as

$$\mu^{\pi_\theta}(s_0, n) := \mathbb{E}\Big[\frac{1}{n}\sum_{t=1}^{n} r_t\Big],$$

where $r_1, r_2, \ldots, r_n$ is the sequence of rewards observed by running the policy $\pi(\cdot|\cdot;\theta)$ from time $t = 0$ to $t = n - 1$ staring at s_0. The random process $(\mu^{\pi_\theta}(s_0, n))_n$ converges to a fixed point, which is independent of initial state s_0, under the assumption that the Markov reward process induced by the policy $\pi \in \mathcal{G}$is ergodic:

$$\mu(\theta) := \lim_{n \to \infty} \mu^{\pi_\theta}(s_0, n),$$

where $\theta \in \Theta$ is the set of parameters which represents the policy $\pi_\theta \in \mathcal{G}$. The goal is to find the best $\theta^* \in \Theta$ which maximizes $\mu(\theta)$, that is, $\theta^* \in \{\arg\max_{\theta \in \Theta} \mu(\theta)\}$. The corresponding best policy is denoted by π^*_Θ. [11]

This setting is a special case of the general scenario considered in Sect. 2. The adaptation of notation and assumptions from Sect. 2 to cover the MDP notation is rather straightforward: the parameter space $\theta \in \Theta$ corresponds to the space of arms $\mathcal{X}$, since in the policy search we want to explore the parameter space Θ to learn the best parameter θ^*. Also the state space $\mathcal{S}$ in MDP setting is the special from of context space of Sect. 2 where here the contexts evolve according to some controlled Markov process. Further the transition kernel T, which at each time step t determines the distribution on the current state and reward given the last state and θ is again a special case of of the more general $(Q_t)_t$ which may depend on the entire history of prior observations. Likewise $\mu(\theta)$, μ^*_Θ and θ^* translate into $f(\theta)$, f^* and x^*, respectively, using

[11] We note that π^*_Θ may be considered optimal only w.r.t. the policies in the policy class $\mathcal{G}$. In general the optimal policy of the MDP, π^*, can be different from π^*_Θ, since $\mathcal{G}$ may not include π^*.

the notation of Sect. 2. The Asm. 1 and 2 in Sect. 2 are also the general version of the standard ergodicity and mixing assumption in MDPs, in which the notion of filtration in assumptions of Sect. 2 is simply replaced by the the initial state $s_0 \square S$.

Based on this adaptation one can simply use *HCT*-Γ algorithm to find the best policy $\pi^*_\Theta \square G$. The advantage of *HCT*-Γ algorithm to prior works in policy search literature is that, to the best of our knowledge, it is the first policy search algorithm which provides finite sample guarantees in the form of regret bounds on the performance loss of policy search in MDPs, as has been proved in Thm.2. This result guarantees that *HCT*-Γ poses a small sub-linear regret w.r.t. π^*_Θ along the way. Also it is not difficult to prove that the policy induced by *HCT*-Γ has a small simple regret, that is, its average reward converges to $\mu(\theta^*)$ with a polynomial rate.[12]

In the context of MDPs, another work somehow related to *HCT*-Γ is the UCCRL algorithm by Ortner & Ryabko (2012), which extends the original UCRL algorithm (Jaksch et al., 2010) to continuous state spaces. Although a direct comparison between the two methods is not possible, it is interesting to notice that the assumptions used in UCCRL are stronger than for *HCT*-Γ, since they require both the dynamics and the reward function to be globally Lipschitz. Furthermore, UCCRL requires the action space to be finite, while *HCT*-Γ can deal with any continuous policy space. Finally, while *HCT*-Γ is guaranteed to minimize the regret against the best policy in the policy class G, UCCRL targets the performance of the actual optimal policy of the MDP at hand.

[12] The reader is referred to Bubeck et al. (2011a); Munos (2013) for details of transforming bounds on accumulated regret to simple regret bounds.