Adaptive-treed bandits

Adam D. Bull
Statistical Laboratory
University of Cambridge

Abstract

We describe a novel algorithm for continuum-armed bandits and noisy global optimisation, with good convergence properties over any continuous reward function having finitely many polynomial maxima. Over such functions, our algorithm achieves square-root cumulative regret in bandits, and inverse-square-root error in optimisation, without prior information.

Our algorithm works by reducing these problems to tree-armed bandits, and we also provide new results in this setting. We show it is possible to adaptively combine multiple trees so as to minimise the regret, and also give near-matching lower bounds on the regret in terms of the zooming dimension.

1 Introduction

Multi-armed bandits are one of the fundamental problems in sequential decision theory. In essence, the problem is to optimally play an unknown slot machine with multiple arms. This problem was originally considered as a model for the sequential design of experiments (Robbins, 1952), but has more recently been of interest in noisy optimisation, artificial intelligence and online services.

In its simplest form, the problem is as follows. At each time step $t$, we choose an arm $x_t$ from an arm space $X = \{1, \ldots, K\}$. We then receive a reward $Y_t \in [0, 1]$, drawn from some unknown distribution $P(x_t)$. Our goal is to choose the arms $x_t$ sequentially, so as to maximise the rewards $Y_t$.

To solve the problem well, we must correctly make a trade-off between exploration and exploitation. At first, we must explore the problem, trying every arm $x_t$ so we can learn the distributions $P$. To succeed, however, we must also exploit that knowledge, preferring those arms $x_t$ which offer the best rewards.

MSC 2010: 62C20 (Primary); 62L20, 68T05, 90C26 (Secondary).

Keywords: continuum-armed bandits, noisy global optimisation, tree-armed bandits, bandits on taxonomies, zooming dimension.
This trade-off between exploration and exploitation is characteristic of many sequential decision problems, and multi-armed bandits have thus been extensively studied (see references in Kleinberg, 2005; Bubeck and Cesa-Bianchi, 2012). Recently, research has also focused on some more complex problems, which share this exploration-exploitation trade-off.

For example, when playing the game of Go, computer programs must investigate a tree of possible strategies. Given limited resources, these programs must divide their time between exploration, trying many different strategies; and exploitation, investigating those strategies which perform well.

Some of the top-ranked Go-playing programs work using bandit techniques, letting the arm space $X$ be given by the set of all possible strategies. As this set is far too large to explore exhaustively, these programs instead use the tree structure of the strategies to search more intelligently (see references in Gelly et al., 2012).

We can think of this as a tree-armed bandit problem: a bandit problem where the arm space $X$ has some tree structure, and arms which are close on the tree offer similar rewards. Such problems occur not only in computer game-playing, but also in a variety of online services (Kocsis and Szepesvári, 2006; Coquelin and Munos, 2007; Pandey et al., 2007; Slivkins, 2011; Yu and Mannor, 2011).

A third example is the problem of noisy global optimisation. Suppose we have a continuous function $\mu : [0,1]^p \rightarrow [0,1]$ we wish to maximise, given only noisy observations of the function values $\mu(x_t)$. A good solution to this problem must make the correct trade-off between exploration, looking for new local maxima; and exploitation, investigating those maxima which have already been found.

This problem arises in a wide variety of engineering applications, and has been considered by many authors (for example, see references in Parsopoulos and Vrahatis, 2002; Huang et al., 2006; Frazier et al., 2009; Srinivas et al., 2010). While there are many possible approaches to the problem, we will consider it in a bandit setting; doing so will allow us to construct algorithms which achieve minimax rates of convergence, while still exhibiting good average-case behaviour.

We can think of noisy global optimisation as a continuum-armed bandit problem: a bandit problem where the arm space $X = [0,1]^p$, and the rewards depend continuously on the arms. This problem can in turn be seen as a special-case of tree-armed bandits, and developments can lead to new approaches not only in noisy global optimisation, but also more generally in complex bandit problems such as computer Go.

In the following paper, we will provide new methods and analysis for continuum-armed bandits, by first reducing to a tree-armed bandit problem. As a consequence, we will describe a new algorithm for noisy global optimisation, improving upon previous bandit algorithms in the literature,
as well as new results for tree-armed bandits, which may be of wider interest.

We now give some further definitions, discuss previous work, and then outline our contributions. Many results on bandit problems are stated in terms of the reward function

\[ \mu(x) = \mathbb{E}[Y_t \mid x_t = x], \]

and aim to minimise the cumulative regret,

\[ R_T = \sum_{t=1}^{T} (\mu^* - \mu(x_t)), \]

where \( \mu^* = \sup_{x \in X} \mu(x) \).

If our goal is to optimise the function \( \mu \), at time \( T \) we may additionally return an estimate \( \hat{x}_T \in X \) of a global maximum of \( \mu \), and then minimise the simple regret,

\[ S_T = \mu^* - \mu(\hat{x}_T). \]

If an algorithm controls the cumulative regret at rate \( Tr_T \), it can also control the simple regret at rate \( r_T \) (Bubeck et al., 2009). Controlling the cumulative regret is thus a more powerful result.

From a minimax perspective, in optimisation we may care only about the simple regret. However, we can expect an algorithm which also controls the cumulative regret to have better average-case performance, as it will be additionally required to place most of its observations near global maxima of \( \mu \). In the following, we will therefore focus on algorithms which control the cumulative regret.

The continuum-armed bandit problem was devised by Agrawal (1995), and for Lipschitz reward functions \( \mu \), nearly tight bounds on the cumulative regret were first proved by Kleinberg (2005). Kleinberg applied the UCB1 strategy of Auer et al. (2002) to a simple fixed discretisation of the arm space \([0, 1]\), achieving \( \tilde{O}(T^{2/3}) \) regret.

Independently, Cope (2009) found it was possible to achieve \( O(\sqrt{T}) \) regret given stronger assumptions on \( \mu \): Cope showed this for the stochastic approximation algorithm of Kiefer and Wolfowitz (1952), applied to unimodal reward functions \( \mu \). Auer et al. (2007) obtained similar bounds by extending the method of Kleinberg (2005): Auer et al. obtained \( \tilde{O}(\sqrt{T}) \) regret over any reward function \( \mu \) with, say, finitely many quadratic global maxima.

Kleinberg et al. (2008) described a new ‘zooming’ algorithm, which used an adaptive discretisation of the arm space \( X \), and could be applied whenever \( X \) was a metric space. For Lipschitz \( \mu \), Kleinberg et al. obtained regret like \( \tilde{O}(T^{1-1/(\beta+2)}) \), for a parameter \( \beta \geq 0 \) they called the zooming dimension, which measures the difficulty of the bandit problem.
Bubeck et al. (2011b) described an algorithm HOO, which improved upon the zooming algorithm of Kleinberg et al. (2008), weakening the assumptions required. Bubeck et al. thereby again obtained $\tilde{O}(\sqrt{T})$ regret over $\mu$ with, say, finitely many quadratic global maxima, as well as covering more general arm spaces and reward functions.

While the above results are significant, a shared weakness is that they all require some assumptions on the shape of the reward function $\mu$. For example, if we assume that $\mu$ has quadratic global maxima, like
\begin{equation}
\mu(x) = -x^2, \tag{4}
\end{equation}
but then try to optimise a maximum of a different power,
\begin{equation}
\mu(x) = -x^4, \tag{5}
\end{equation}
or one which mixes powers,
\begin{equation}
\mu(x, y) = -x^2 - y^4, \tag{6}
\end{equation}
we will not achieve $\tilde{O}(\sqrt{T})$ regret.

Some authors have tried to improve upon this, constructing bandit algorithms which adapt to the shape of the reward function. Under further regularity assumptions, Bubeck et al. (2011a) extended the algorithm of Kleinberg (2005) to adapt to the Lipschitz constant. In a noiseless problem, for the simple regret, Munos (2011) described an algorithm based on HOO, which adapts to a wide range of reward functions $\mu$.

In this paper, we will build upon an approach described by Slivkins (2011) for tree-armed bandits. Slivkins described an algorithm, Taxonomy-Zoom, which is based on the zooming algorithm of Kleinberg et al. (2008), and can adapt to a wide range of reward functions $\mu$, if the arm space $X$ is given by a finite tree.

In the following, we will describe an algorithm based on TaxonomyZoom, which applies more generally in tree-armed bandits. We will prove our algorithm adapts to a wider range of reward functions, using a novel argument which corrects a flaw in the argument of Slivkins (2011).

As a consequence, we will deduce an algorithm for continuum-armed bandits which achieves $\tilde{O}(\sqrt{T})$ cumulative regret over reward functions like (4)–(6); in noisy global optimisation, our algorithm will likewise optimise such functions with error $\tilde{O}(1/\sqrt{T})$. While the constants in these rates will of course depend on the reward function, our algorithm will be able to attain the rates without prior knowledge of the rewards.

In tree-armed bandits, our results will show it is possible for a bandit algorithm to adaptively combine multiple trees over $X$, so as to minimise the regret. We will also prove near-matching lower bounds on the regret, given in terms of the zooming dimension $\beta$. 

4
Concurrently with this work, Valko et al. (2013) have described another adaptive algorithm which can be applied to continuum-armed bandits, based on the approach of Munos (2011). While their results only bound the simple regret, and do not adapt to reward functions like (6), their approach may be more easy to generalise, and their results are complementary to ours.

In Section 2, we will discuss the continuum-armed bandit problem, and the class of reward functions \( \mu \) we consider. In Section 3, we will then describe our algorithm for tree-armed bandits, and state the results we will prove in this setting. Finally, in Section 4 we will give proofs of our results.

2 Continuum-armed bandits

In this section, we will describe our results on the continuum-armed bandit problem. In Section 2.1, we will give a definition of the problem, and in Section 2.2, define the class of reward functions we will consider. In Section 2.3 we will give examples of our new function class, and in Section 2.4 briefly state our results.

2.1 Problem statement

We begin with a precise definition of the multi-armed bandit problems we will consider. Suppose we have some measurable arm space \((X, \mathcal{E})\), and for each \( x \in X \), we have an unknown distribution \( P(x) \) over \([0, 1] \). At time \( t \), we are allowed to choose an arm \( x_t \in X \), and then receive a reward \( Y_t \) with distribution \( P(x_t) \), which we wish to maximise.

Formally, we take a probability space \((\Omega, \mathcal{F}, \mathbb{P})\), with random variables \( Y_t \in [0, 1] \) and \( Z_t \in \mathbb{Z} \), \( t \in \mathbb{N} \), for a measurable space \((\mathbb{Z}, \mathcal{E}')\); the variables \( Z_t \) represent a source of randomisation. At time \( t \), we require that \( Z_t \) is distributed independently of past events, \( x_t \) is an \( \mathcal{E} \)-measurable function of \( Y_1, \ldots, Y_{t-1}, Z_1, \ldots, Z_t \), and \( Y_t \) has distribution \( P(x_t) \), conditionally on past events and \( Z_t \). A strategy for the continuum-armed bandit problem is given by the functions \( x_t \), and the distributions of the variables \( Z_t \).

We can then define the reward function \( \mu \) and cumulative regret \( R_T \), as in (1) and (2). If our goal is to optimise \( \mu \), we may additionally return an estimated global maximum \( \hat{x}_T \), which we require to be an \( \mathcal{E} \)-measurable function of \( Y_1, \ldots, Y_T, Z_1, \ldots, Z_T \), and an independent randomisation variable \( \hat{Z}_T \in \mathbb{Z} \). We can then likewise define the simple regret \( S_T \), as in (3).

Our goal is to find a strategy which makes the regrets \( R_T \) and \( S_T \) as small as possible, for a wide variety of reward functions \( \mu \). If the space \( X \) is finite, this is the standard multi-armed bandit problem. In the following, however, we will first consider the space \( X = [0, 1]^p \), for \( p \in \mathbb{N} \); this is then the continuum-armed bandit problem, illustrated in Figure 1.

As noted in the introduction, previous works have been able to establish cumulative regret like \( \tilde{O}(\sqrt{T}) \), and hence also simple regret like \( \tilde{O}(1/\sqrt{T}) \),
Figure 1: The continuum-armed bandit problem: we choose arms $x_t$, observe rewards $Y_t$, and aim to maximise the mean rewards $\mu(x_t)$.

only under strong assumptions on the shape of $\mu$ near its global maxima. In the following, our goal will be to achieve similar results, for more general reward functions, without prior knowledge of $\mu$.

2.2 Zooming continuous functions

We now define a new class of reward functions $\mu$, the zooming continuous functions. We will show that this class covers a wide variety of potential behaviours for the reward function $\mu$, and that it is possible to solve any such continuum-armed bandit problem with only $\tilde{O}(\sqrt{T})$ cumulative regret.

The motivation for our definition comes from the behaviour of zooming-type algorithms for continuum-armed bandits, such as the zooming algorithm of Kleinberg et al. (2008), or the HOO algorithm of Bubeck et al. (2011b). We note that these algorithms proceed by ‘zooming in’ to small regions $U$ of the arm space $X$, in neighbourhoods of the global maxima of $\mu$.

These algorithms choose arms $x_t$ preferentially from the regions $U$, attempting to learn more about the behaviour of $\mu$ near its maxima. Given enough data, the algorithms then subdivide each region $U$ into smaller regions, and focus on those closest to the maxima of $\mu$.

The regret incurred by such algorithms in large part depends on the difficulty of finding sub-regions of $U$ on which $\mu$ varies less. We therefore wish to find a wide class of functions $\mu$ for which this task is not too difficult.

The smoother $\mu$ is, the less $\mu$ will vary near its maxima, and the easier this task will be. We are therefore looking for some kind of smoothness class of functions. However, in this case standard smoothness classes do not describe the correct behaviour, as we will show in Section 2.3.
It turns out that the relevant smoothness class is a kind of uniform continuity, which we call zooming continuity. The definition is as follows. Given any $U \subseteq \mathbb{R}^p$, define its diameter along axis $i$,

$$\text{diam}_i(U) = \sup\{|x_i - y_i| : x, y \in U\},$$

and its overall diameter,

$$\text{diam}(U) = \sup\{\|x - y\| : x, y \in U\}.$$

Given also $x \in \mathbb{R}^d$, define its size, relative to $U$, to be

$$\|x\|_U^2 = \sum_{i=1}^p \left(\frac{|x_i|}{\text{diam}_i(U)}\right)^2.$$

We can then state the following definition.

**Definition 1.** Let $X \subset \mathbb{R}^p$ be a compact product of intervals. The function $f : X \to \mathbb{R}$ is zooming continuous if:

(i) $f$ is continuous, with finitely many global maxima; and

(ii) for any global maximum $x^*$ of $f$, and any neighbourhood $U$ in $X$ of $x^*$,

$$\sup_{x^*, U : \text{diam}(U) \leq \epsilon} \sup_{x, y \in U : \|x - y\| \leq \epsilon} \frac{|f(x) - f(y)|}{\sup_{z \in U} |f(x^*) - f(z)|} \to 0$$

as $\epsilon \to 0$.

Intuitively, we are describing functions $f$ with the following property. Let $U$ be a neighbourhood in $X$ of a global maximum $x^*$. Suppose we restrict $f$ to $U$, and then rescale $f$ so that its domain $U$ is just contained within $[0, 1]^p$, and its range over $U$ is $[0, 1]$. In other words, we ‘zoom in’ to $f$ over $U$.

This new, zoomed-in $f$ should be uniformly continuous. Furthermore, its continuity should be uniform over all choices of $U$ and $x^*$: the function $f$ should not get rougher as we zoom in on smaller and smaller regions $U$.

Our definition is illustrated in Figure 2. Function (i) looks zooming continuous: as we zoom in on its maximum, the function remains smooth. Function (ii) does not look zooming continuous: as we zoom in on its maximum, the function gets rougher.

### 2.3 Examples of zooming continuity

We now explore the consequences of our definition, by considering some simple examples of functions which are or are not zooming continuous.

**Example 1.** Let $f : [-1, 1] \to \mathbb{R}$. For any $\alpha > 0$, the following function $f$ is zooming continuous.
Figure 2: Function (i) looks zooming continuous; (ii) does not.

\( f(x) = -|x|^\alpha. \)

However, the following functions \( f \) are not zooming continuous.

(ii) \( f(x) = 1/ \log(1/2x^2). \)

(iii) \( f(x) = -\exp(-1/x^2). \)

(iv) \( f(x) = -|x|^{1+1(x>0)}. \)

The above functions are illustrated in Figure 3. We note that maxima given by powers of \(|x|\) are thus zooming continuous; however, some other elementary functions are not.

The functions which fail to satisfy Definition 1 are all difficult to handle within the framework we will describe, for various reasons. Intuitively, it is unsurprising we do not consider function (ii) of Example 1, as it has a sharply peaked maximum, which will prove difficult to locate under noise.

Function (iii), which has a very flat maximum, and (iv), which looks like two different polynomials, should be easier to optimise. However, they are difficult to include in the technical framework we will introduce below; a more sophisticated analysis may be able to improve upon this. Similar problems also occur when considering functions with infinitely many global maxima, which we have excluded from Definition 1.

Nevertheless, the functions (i) still cover a wide range of behaviour for reward functions in continuum-armed bandits. In Section 2.4, we will show that zooming continuity thus gives a more general class of reward functions than has been previously considered when establishing good rates of cumulative regret.
Figure 3: Functions (i)–(iii) are zooming continuous; (iv)–(vi) are not.

First, though, we demonstrate that zooming continuity is independent of some standard smoothness classes. For $s > 0$, let $C^s$ denote the class of $s$-Hölder functions, and let $C^\infty$ denote the class of infinitely-differentiable functions. As the functions $-|x|^\alpha$ are not $C^s$ for any $s > \alpha$, but the function $-\exp(-1/x^2)$ is $C^\infty$, we have the following corollary.

**Corollary 1.** Let $f : [-1, 1] \to \mathbb{R}$.

(i) For any $s > 0$, there exist zooming continuous functions $f$ which are not $C^s$.

(ii) There exist functions $f$ which are $C^\infty$, but are not zooming continuous.

### 2.4 Results for continuum-armed bandits

We now move on to our results on the regret in continuum-armed bandits, for zooming continuous reward functions. We begin with some general examples of zooming continuous functions, which will be more relevant to the problem we consider.

**Example 2.** Let $X \subset \mathbb{R}^p$ be a compact product of intervals, and $f : X \to \mathbb{R}$ be continuous, with finitely many global maxima $x_1^*, \ldots, x_L^*$. For each maximum $x_j^*$, let $f$ satisfy one of the following as $x \to x_j^*$.

(i) $x_j^*$ is an elliptical maximum,

$$f(x) = f(x_j^*) - ||A_l(x - x_j^*)||^{\alpha_l}(1 + o(1)),$$

for a positive-definite matrix $A_l \in \mathbb{R}^{p \times p}$, and $\alpha_l > 0$. 

(ii) $x_1^*$ is a separable maximum,

\[ f(x) = f(x_1^*) - \left( \sum_{i=1}^{p} c_{l,i}|x_i - x_{l,i}^*|^{\alpha_{l,i}} \right) (1 + o(1)), \]

for $c_{l,i}, \alpha_{l,i} > 0$.

Then $f$ is zooming continuous.

We note that the case of elliptical maxima includes all maxima where the function is well-approximated by a quadratic, setting $\alpha_{l} = 2$, and letting $A_l$ be the square root of the Hessian matrix. Alternatively, the case of separable maxima allows us to model functions which depend more strongly on some coordinates $x_i$ than others.

In general, we can think of zooming continuity as describing reward functions with polynomial-like behaviour at their global maxima. If these maxima were known to be quadratic, say, then under some additional assumptions, previous algorithms for continuum-armed bandits, such as the UCBC algorithm of Auer et al. (2007), or the HOO algorithm of Bubeck et al. (2011b), would be able to achieve $\tilde{O}(\sqrt{T})$ cumulative regret.

However, our zooming continuity condition is more general. It describes the behaviour of reward functions with polynomial-like global maxima, for all polynomial powers simultaneously; it even allows these powers to be different along each axis.

When the reward function has such behaviour, the following result shows we can achieve good rates of regret in the continuum-armed bandit problem. This result comes as a corollary to theorems given in Section 3, where we also describe a strategy achieving such regret, and provide a near-matching lower bound.

**Corollary 2.** Let $\varepsilon \in (0, 1)$. There exists a strategy for continuum-armed bandits, depending only on $\varepsilon$, which achieves regret

\[ R_T = \tilde{O}(\sqrt{T}), \quad S_T = \tilde{O}(1/\sqrt{T}), \]

on an event with probability $1 - \varepsilon$, whenever the reward function $\mu$ is zooming continuous. Furthermore, on this event, the strategy has a total computation time of only $\tilde{O}(T)$.

We thus deduce that it is possible to achieve $\tilde{O}(\sqrt{T})$ cumulative regret in continuum-armed bandits, and $\tilde{O}(1/\sqrt{T})$ error in noisy global optimisation, over a wide variety of reward functions, without prior information. However, to establish this result, we must first reduce the problem to one of tree-armed bandits, which we consider in the following section.
3 Tree-armed bandits

In this section, we will describe our results on the tree-armed bandit problem. In Section 3.1, we will give a definition of the problem, and in Section 3.2, describe the algorithm we will use to solve it. In Section 3.3, we will define a class of reward functions over which our algorithm performs well, and in Section 3.4, state our bounds on the regret. Finally, in Section 3.5 we will show how our algorithm can be implemented efficiently.

3.1 Problem statement

In the tree-armed bandit problem, we again consider the multi-armed bandit problem described in Section 2.1, but now with a more general arm space $X$. We allow any space $X$ on which we are given a certain tree structure, which we define below; we will show that the continuum-armed bandit problem is a special-case of this more general setting.

We choose to work in the tree-armed setting, rather than the continuum-armed one, for a number of reasons. Firstly, in the tree-armed setting our conditions on the reward function $\mu$ are more easily relatable to those studied by previous authors, including the zooming-dimension condition of Kleinberg et al. (2008), and the quality condition of Slivkins (2011).

Secondly, the tree-armed setting allows us to state our results in more generality; in particular, we prove near-matching upper and lower bounds for reward functions of any zooming dimension, a result which to our knowledge is new to the literature. Finally, the tree-armed setting also makes it easier to state and prove our results for continuum-armed bandits.

The tree-armed bandit problem has been independently considered by several authors (Kocsis and Szepesvári, 2006; Coquelin and Munos, 2007; Pandey et al., 2007; Slivkins, 2011; Yu and Mannor, 2011), and the strategies developed are currently used by some of the top-ranked computer programs in the game of Go (Gelly et al., 2012). While our results will be motivated by the specific problem of continuum-armed bandits, they may thus also be of wider interest.

To define our setting, let the arm space $X$ be a Cartesian product $\prod_{i=1}^{p} X_i$, for coordinate spaces $X_i$. For $i = 1, \ldots, p$, let $T_i$ be a tree with root node $X_i$, and whose nodes are all given by non-empty subsets of $X_i$. Further require that each node $U$ is either a leaf node, or has children $V$ which form a partition of the set $U$. Each non-leaf node must have at least 2 and at most $q$ children, for a constant $q \in \mathbb{N}$.

Formally, we can define a $\sigma$-algebra $\mathcal{E}$ on $X$ as follows. For each coordinate space $X_i$, let $\mathcal{E}_i$ be the sigma-algebra generated by the nodes $U$ of $T_i$. We then define $\mathcal{E}$ to be the product $\sigma$-algebra of the $\mathcal{E}_i$.

As before, we sequentially choose arms $x_t \in X$, and receive rewards $Y_t \in [0,1]$ having distribution $P(x_t)$. Our goal remains to find a strategy
minimising the regrets $R_T$ and $S_T$, for a wide variety of reward functions $\mu$. However, we must now do so for general treed spaces $X$, given only the trees $T_i$.

Continuum-armed bandits lie within this setting, letting each coordinate space $X_i = [0, 1]$. The trees $T_i$ can be chosen to be dyadic trees on $[0, 1]$, defined as follows. The dyadic tree on $[0, 1)$ is the tree with root node $[0, 1)$, where each node $[a, b)$ has children $[a, \frac{1}{2}(a + b))$, $[\frac{1}{2}(a + b), b)$. We can similarly define the dyadic tree on $[0, 1]$, instead allowing each node with upper bound 1 to contain the point 1; this tree is illustrated in Figure 4. If the trees $T_i$ are dyadic trees on $[0, 1]$, then $\mathcal{E}$ is the Borel $\sigma$-algebra on $[0, 1]^p$, and we recover the continuum-armed bandit setting of Section 2.1.

With these definitions, we can consider continuum-armed bandits as a special-case of tree-armed bandits. In the following section, we will describe an algorithm for solving not only continuum-armed bandits, but also tree-armed bandits more generally. We will then show that our algorithm not only improves upon previous work in tree-armed bandits, but also achieves near-optimal rates of regret over zooming continuous reward functions.

3.2 Adaptive-treed bandits

Our algorithm for tree-armed bandits uses a UCB-like strategy, where the arms $x_t$ are chosen to maximise an upper confidence bound on the reward. Such strategies were popularised by Auer et al. (2002), and have previously been applied to both continuum-armed bandits (Kleinberg, 2005; Auer et al., 2007; Kleinberg et al., 2008; Bubeck et al., 2011b) and tree-armed-bandits (Kocsis and Szepesvári, 2006; Coquelin and Munos, 2007; Pandey et al., 2007; Slivkins, 2011).

In particular, our algorithm builds upon the work of Slivkins (2011), and

Figure 4: The dyadic tree over $[0, 1]$. 

![Diagram of the dyadic tree over [0, 1]](image-url)
is closely related to that author’s TaxonomyZoom algorithm for bandits on a single finite tree. Our algorithm differs from TaxonomyZoom in three respects.

Firstly, our algorithm works with infinite trees. While this change is relatively minor, it is necessary for the application to continuum-armed bandits. Secondly, our algorithm returns an estimated global maximum $\hat{x}_T$, allowing the algorithm to be directly used in optimisation. Finally, and most significantly, our algorithm can work with multiple trees, adaptively combining them so as to minimise the regret.

We begin with a brief summary of the algorithm. At time $t$, we will partition the arm space $X$ into a set $A_{t-1}$ of active boxes, chosen in terms of the past rewards $Y_1, \ldots, Y_{t-1}$. For each box $B \in A_{t-1}$, we further compute an index $I_{t-1}(B) \in \mathbb{R}$, which upper bounds the maximum reward $\sup_{x \in B} \mu(x)$. We will then select an active box $B_t$ maximising $I_{t-1}$, and pull an arm $x_t$ chosen uniformly at random from $B_t$.

To describe our algorithm in detail, we will need some additional definitions. We begin with the concepts which depend on the sample space $X$: the set of boxes $B \subseteq X$ we will use to construct our partitions, and the distribution $\pi$ over $X$ we will treat as uniform.

In the specific case of continuum-armed bandits, the boxes $B$ will be the products of dyadic intervals in $[0, 1]^p$, and $\pi$ will be the uniform distribution on $[0, 1]^p$. However, since we aim to treat the more general tree-armed setting, we now give more general descriptions of these ideas.

We define a box $B$ to be any product $\prod_{i=1}^p U_i$, where each $U_i$ is a node in the tree $T_i$; we further let $B$ denote the set of all such boxes. For a fixed reward function $\mu : X \to [0, 1]$, we also define the width $W$ of a box $B$ to be

$$W(B) = \sup_{x \in B} \mu(x) - \inf_{x \in B} \mu(x).$$

We then define a distribution $\pi$ on the measurable space $(X, \mathcal{E})$, given as the product of distributions $\pi_i$ on the spaces $(X_i, \mathcal{E}_i)$. Intuitively, $\pi_i$ will be the distribution of a point in $X_i$ chosen by uniform random descent of the tree $T_i$.

To be precise, we generate a random sequence of nodes $U_n$ in $T_i$, setting $U_1 = X_i$. For $n \in \mathbb{N}$, if $U_n$ is a leaf node, we terminate the sequence at $U_n$; otherwise, we choose $U_{n+1}$ uniformly at random from the children of $U_n$. We then define a distribution $\pi_i$ on $(X_i, \mathcal{E}_i)$ by

$$\pi_i(U) = \mathbb{P}(\exists n \in \mathbb{N} : U_n = U), \quad U \in T_i;$$

it can be checked this uniquely defines a distribution $\pi_i$.

We have thus defined the set $B$ of boxes we will use to partition $X$, and the distribution $\pi$ over $X$ we will take as uniform. We note that for continuum-armed bandits, these definitions agree with those given above.
We now move onto the definition of the index $I_t$. For each active box $B$, $I_t(B)$ will be based on the empirical mean of past rewards $Y_s$ associated with arms $x_s$ in $B$. To ensure this is an upper bound for the maximum reward over $B$, we will add two additional terms: one to correct for the stochastic error associated with estimating the mean reward, and one to bound the difference between mean and maximum.

Suppose that at time $t$, we select the active box $B_t$, drawing $x_t$ from the distribution $\pi | B_t$. For any box $B$, we will say $B$ was hit at time $t$ if $x_t \in B \subseteq B_t$. Let $n_t(B)$ be the number of times $s \leq t$ at which $B$ was hit, and if $n_t(B) > 0$, let $\mu_t(B)$ be the corresponding average reward. For fixed $\mu$, we note that $\mu_t(B)$ is an unbiased estimate of $\mu(B) = \mathbb{E}_\pi[\mu(x) | x \in B]$, the expected reward on $B$ under $\pi$.

To bound the error in this estimate, we next define a confidence radius $r_t(B)$, chosen so that $|\mu_t(B) - \mu(B)| \leq r_t(B)$ with high probability. We first fix an error probability $\varepsilon \in (0, 1)$, which will control the accuracy of our bound; we will show that our results on the regret hold with probability $1 - \varepsilon$.

For any box $B = \prod_{i=1}^p U_i$, we then let $d(B)$ denote the depth of $B$, the maximum depth of any $U_i$ in its corresponding tree $T_i$, and define the constant $\rho(B) = q^{\rho(d(B)+1)}$.

We also set $\tau = 4\varepsilon^{-1}$, and then define the confidence radius

$$r_t(B) = 2\sqrt{\log[\rho(B)(\tau + n_t(B))]}/n_t(B).$$

To conclude the definition of the index $I_t(B)$, we will need a term bounding the difference between the mean and maximum reward on $B$. This term will depend on a constant $\gamma \in (0, 1)$, which we will call the quality, a notion we inherit from Slivkins (2011).

The quality $\gamma$ describes how difficult we expect the tree-armed bandit problem to be, and thus how conservatively our algorithm should act. In the following sections, we discuss the implications of $\gamma$ in more detail. For now, we note that smaller $\gamma$ corresponds to a more difficult problem, and more conservative behaviour.

Given a fixed choice of $\gamma$, we then define the index

$$I_t(B) = \mu_t(B) + (1 + 2p\nu)r_t(B),$$

where the constant $\nu = 8\sqrt{2/\gamma}$; if $n_t(B) = 0$, we take $I_t(B) = +\infty$. The index $I_t(B)$ is thus a sum of the empirical mean $\mu_t(B)$, the confidence radius $r_t(B)$, and an additional term $2p\nu r_t(B)$, which bounds the difference
between the mean and maximum reward over $B$; this is illustrated in Figure 5.

We next describe our set $A_t$ of active boxes. Our goal will be to choose as few active boxes as possible, while still ensuring that for each active box $B$, the index $I_t(B)$ is an upper bound for the maximum reward over $B$. To do so, we will aim to select a set of active boxes $B$ satisfying the inequality $W(B) \leq 2\rho \nu r_t(B)$; we will thus need to find estimates of the widths $W(B)$.

Our estimates will work on the principle that, if the reward function $\mu$ is well-behaved, we will be able to find large enough sub-boxes $C_1, C_2 \subseteq B$ for which $\mu(C_1) - \mu(C_2) \approx W(B)$. Since we always have $\mu(C_1) - \mu(C_2) \leq W(B)$, we may thus estimate $W(B)$ by a suitable maximum of these differences, taken over many pairs $C_1, C_2$.

Since we will not have access to the means $\mu(C_k)$ themselves, we will need to bound them using the data. We therefore define the lower and upper bounds on the mean reward,

$$\underline{\mu}_t(B) = \mu_t(B) - r_t(B), \quad \overline{\mu}_t(B) = \mu_t(B) + r_t(B).$$

We may then define our width estimate

$$W_t(B) = \max_{(C_1, C_2) \in \mathcal{M}_t(B)} \underline{\mu}_t(C_1) - \overline{\mu}_t(C_2),$$

where the maximum is taken over a set $\mathcal{M}_t(B)$ of pairs $(C_1, C_2)$ to be defined; the estimate $W_t(B)$ is illustrated in Figure 6.

We will choose $\mathcal{M}_t(B)$ to be the set of all pairs $(C_1, C_2)$ of boxes $C_1, C_2 \subseteq B$, which for $k = 1, 2$ satisfy:

(i) $\pi(C_k \mid B) \geq \gamma$; and
We have thus described how we choose the set \( \mathcal{A}_t \) of active boxes. Finally, we define our estimate \( \hat{x}_T \) of a global maximum of \( \mu \); we set \( \hat{x}_T = x_{T^*} \), where the optimal time

\[
T^* = \arg \min_{t=1}^T r_t(B_t),
\]

breaking ties arbitrarily.

We have then described in full our algorithm ATB, given by Algorithm 1. We note that our algorithm is closely related to the Taxonomy Zoom algorithm of Slivkins (2011); we briefly describe the changes.
Figure 7: Plot (i) shows a partition $\mathcal{A}_t$ of the arm space $X = [0,1]^2$ into active boxes; (ii) shows $\mathcal{A}_t$ after the box $B$ has been split to maintain Invariant 1. The boxes $C_1, C_2$ satisfy condition (9).

Data: space $X$, trees $T_i$, error rate $\varepsilon$, quality $\gamma$

set $\mathcal{A}_0 = \{X\}$;

for $t = 1, \ldots, T$ do

select a box $B_t \in \mathcal{A}_{t-1}$ maximising $I_{t-1}$;
play an arm $x_t$ drawn at random from $\pi | B_t$;
set $\mathcal{A}_t = \mathcal{A}_{t-1}$;

while Invariant 1 is violated, by $B = \prod_{j=1}^{p} U_j$, and $C_1, C_2$
differing along axis $i$ do

remove $B$ from $\mathcal{A}_t$;

for $V$ a child of $U_i$ in $T_i$ do

add $U_1 \times \cdots \times U_{i-1} \times V \times U_{i+1} \times \cdots \times U_p$ to $\mathcal{A}_t$;

end

end

return $x_{T^*}$

Algorithm 1: Adaptive-treed bandits (ATB)
Firstly, to allow us to work with infinite trees, we have redefined the confidence radius $r_t(B)$ to depend on the depth of $B$, rather than the size of the trees $T_i$. Secondly, we have included a rule for choosing an optimal point $\hat{x}_T$, given in terms of the confidence radii $r_t(B_t)$. Lastly, we have made a number of changes which allow us to work with multiple trees $T_i$.

The first of these is simply that we partition the arm space $X$ into boxes $B \in B$ given by a product of nodes in trees, rather than the nodes themselves. The second is that we have altered the width estimate $W_t(B)$ to require that the boxes $C_1, C_2$ agree except in one axis; this allows us to detect not only the width of a box $B$, but also an axis $i$ along which it varies.

The final change is in the procedure for ensuring that Invariant 1 holds. When the invariant is violated by a box $B$, we split that box only along the axis $i$; this process allows us to adapt the shape of the active boxes $B$ to the shape of the reward function $\mu$.

In Section 3.4, we will show that for a wide range of reward functions $\mu$, our algorithm achieves near-optimal rates of regret, for the optimal combination of trees $T_i$, without prior knowledge of $\mu$. It thereby obtains $\tilde{O}(\sqrt{T})$ cumulative regret in continuum-armed bandits, and $\tilde{O}(1/\sqrt{T})$ error in noisy global optimisation, over any zooming continuous reward function.

### 3.3 Well-behaved rewards

We now describe the conditions we will require on the reward function $\mu$, so as to achieve good rates of regret in tree-armed bandits. As our conditions will be quite technical, we will motivate them in two different ways.

Firstly, our conditions will all be satisfied by zooming continuous reward functions $\mu$; our results will thus apply directly to continuum-armed bandits and noisy global optimisation. Secondly, our conditions will be closely related to other conditions studied in the literature, including the zooming-dimension condition of Kleinberg et al. (2008), and the quality condition of Slivkins (2011).

To begin, we will need some preliminary definitions. In the following, we will consider collections $C$ of pairwise-disjoint boxes $B \in B$. We will say a box $B$ is on $C$, if it is a union of boxes in $B$. We will further say $C$ is a refinement of $C'$, if this is true for all $B \in C'$.

A specific type of collection $C$ we will consider is the grid. A grid $G$ is any set of boxes $\{\prod_{i=1}^p U_i : U_i \in S_i\}$, where for each $i = 1, \ldots, p$, $S_i$ is a collection of pairwise-disjoint nodes in $T_i$. We will say two grids $G, G'$ are separated, if for any box $B$ on $G \cup G'$, $B$ is on either $G$ or $G'$.

Finally, for a fixed reward function $\mu$, given $\delta > 0$ we define the level set

$$X_\delta = \{x \in X : \mu^* - \mu(x) \leq \delta\},$$

and for any box $B$, we define its maximum and average badness,

$$\delta(B) = \mu^* - \min_{x \in B} \mu(x), \quad \Delta(B) = \mu^* - \mu(B). \quad (10)$$
Figure 8: A partition $B_m$ of the arm space $X = [0,1]^2$, together with the cover $C_m$, and grids $\mathcal{G}_{t,m}$.

We are now ready to state our conditions on $\mu$.

**Definition 2.** Let $\mu : X \to [0,1]$ be $\mathcal{F}$-measurable. We will say $\mu$ is well-behaved if for each $m \in \mathbb{N}$, we have a partition $B_m$ of $X$, made up of boxes $B \in \mathcal{B}$, and a subset $C_m \subseteq B_m$, satisfying the following.

(i) For each $m \in \mathbb{N}$, letting $\delta_m = 2^{1-m}$, the level set $X_{\delta_m}$ is covered by $C_m$.

(ii) Each $C_m$ has cardinality at most $\kappa \delta_m^{-\beta}$, for constants $\kappa > 0$, $\beta \geq 0$.

(iii) For each $m \in \mathbb{N}$, the boxes $B \in C_m$ satisfy:

(a) $W(B) \leq \delta_m/12p$, and
(b) $d(B) \leq \lambda m$, for a constant $\lambda > 0$.

(iv) For each box $B$ on some $C_m$, there exist two sub-boxes $C_1, C_2 \subseteq B$ satisfying condition (9), with:

(a) $\pi(C_k \mid B) \geq \gamma$, $k = 1,2$, for a constant $\gamma \in (0,1)$, and
(b) $\mu(C_1) - \mu(C_2) \geq \frac{1}{p}(W(B) - \frac{1}{2}\delta(B))$.

(v) For each $m \in \mathbb{N}$, we have some $L_m \in \mathbb{N}$, and pairwise-separated grids $\mathcal{G}_{1,m}, \ldots, \mathcal{G}_{L_m,m}$, such that $C_m \subseteq \bigcup_{l=1}^{L_m} \mathcal{G}_{l,m}$.

(vi) Each $B_{m+1}$ is a refinement of $B_m$.

We will call $\beta$ the zooming dimension, and $\gamma$ the quality.

We now discuss the implications of our definition, which is illustrated in Figure 8. Firstly, we note that the conditions are all satisfied when the reward function $\mu$ is zooming continuous.
Theorem 1. Let the arm space $X = [0, 1]^p$, given as the product of coordinate spaces $X_i = [0, 1]$, $i = 1 \ldots , p$, with dyadic trees $T_i$ over each $X_i$. If $\mu : X \to [0, 1]$ is zooming continuous, then $\mu$ is well-behaved, with zooming dimension $\beta = 0$.

Secondly, we note that the conditions of Definition 2 are related to other conditions previously studied in the literature. The zooming dimension $\beta \geq 0$, and quality $\gamma \in (0, 1)$, are related to similar concepts defined by Kleinberg et al. (2008) and Slivkins (2011), and measure the difficulty of solving a bandit problem with reward function $\mu$, when subdividing the arm space $X$ using the trees $T_i$.

We will discuss in more detail the meaning of these quantities below; for now, we note that they are a function both of the reward function $\mu$, and the trees $T_i$. In the following, we will assume that we have some natural choice of trees $T_i$ we may treat as fixed, as is the case in continuum-armed bandits; we may thus consider these quantities primarily as a function of $\mu$.

Conditions (i)—(iiia) state that $\mu$ has zooming dimension $\beta \geq 0$: this concept was introduced by Kleinberg et al. (2008), and bounds the number of near-maximal boxes $B$ we must evaluate to find the global maxima of $\mu$. The larger $\beta$ is, the more alternatives we must consider, and the worse our regret rates will be.

Kleinberg et al. (2008) defined the zooming dimension relative to a fixed metric, with respect to which $\mu$ is assumed to be Lipschitz. Our formulation is more closely related to that of Slivkins (2011), who did not fix a metric, but instead used the strongest metric which $\mu$ is Lipschitz with respect to.

Our condition improves upon that of Slivkins (2011) by allowing the cover $C_m$ to be made up of boxes $B \in B$, constructed not just from a single tree $T_i$, but also from arbitrary combinations of them. This flexibility allows us to ensure that a wider variety of reward functions $\mu$ will have zooming dimension $\beta = 0$; in particular, it is necessary to get near-optimal rates for the separable maxima in Example 2.

For the continuum-armed bandit problems we will consider, we will always have zooming dimension $\beta = 0$. However, in tree-armed bandits, we will also consider the case $\beta > 0$, as this allows our results to hold in more generality. In particular, we will prove near-matching lower bounds on the regret in terms of all $\beta \geq 0$, a result we believe to be new in the literature.

Condition (iiib) controls the depth of near-maximal boxes $B$: assuming this condition allows us to construct an algorithm which is more computationally efficient. A similar approach is considered by Bubeck et al. (2011b), who discuss artificially truncating trees at a certain depth.

Condition (iv) states that $\mu$ has quality $\gamma \in (0, 1)$: this concept was introduced by Slivkins (2011), and bounds the difficulty in estimating the widths $W(B)$. Our version of this condition is new, and improves upon Slivkins’ in two ways.
Firstly, we require the bound to hold for a larger collection of boxes $B$; we will show this change allows us to correct a flaw in the argument of Slivkins (2011). Secondly, we require the boxes $C_1, C_2$ to satisfy condition (9). In the case $p = 1$, when we have a single tree over $X$, this condition is trivial. However, when $p > 1$, it allows our algorithm to detect the axes along which $\mu$ varies, and so adaptively combine the trees $\mathcal{T}_i$.

Conditions (v) and (vi) are new to this work, and are also required to work with multiple trees efficiently. Again, when $p = 1$ the conditions can be shown to be trivially satisfiable; when $p > 1$, they will be necessary to prove our new results.

Condition (v) requires that the near-maximal boxes $B$ lie within a grid structure; that the boxes can be created by independent subdivisions of the axes $X_i$. This condition will be necessary to ensure that when our algorithm subdivides the axes, it does not create too many active boxes.

Condition (vi) requires that the near-maximal boxes $B \in C_m$ become smaller as $m$ increases: that they describe consistent regions of the arm space $X$ as $\delta_m \to 0$. This condition will be necessary to ensure that as our algorithm progresses, the active boxes created at earlier time steps do not hinder us at later ones.

While the main motivation behind Definition 2 is our application to continuum-armed bandits, our results can also be applied to other tree-armed bandit problems, including those with finite trees. We note that while our definitions do not require it, it will be easiest to satisfy Definition 2 when all leaf nodes $U_i$ in trees $\mathcal{T}_i$ are singleton sets, a condition which should be satisfied by any reasonable choice of trees $\mathcal{T}_i$.

We have thus defined the conditions we will require on our reward functions $\mu$. In the following section, we will show that, given these conditions, Algorithm 1 achieves good rates of regret.

3.4 Results for tree-armed bandits

We now prove our regret bounds for tree-armed bandits. We will prove our results uniformly over a class of reward functions $\mu$, which we describe below.

For an arm space $X$, given as the product of coordinate spaces $X_i$, $i = 1, \ldots, p$, each equipped with tree $\mathcal{T}_i$, a zooming dimension $\beta \geq 0$, a quality $\gamma \in (0, 1)$, and constants $\kappa, \lambda > 0$, let

$$\mathcal{P} = \mathcal{P}(X, T, \beta, \gamma, \kappa, \lambda)$$

denote the class of arm distributions $P$ whose reward functions $\mu$ are well-behaved, with the above constants. We note that the class $\mathcal{P}$ is increasing in the parameters $\beta, \kappa$ and $\lambda$, and decreasing in $\gamma$.

We also fix some notation we will use to describe our rates of regret. Given functions $f, g : \mathbb{N} \to \mathbb{R}$ satisfying $f(T) = O(g(T))$ as $T \to \infty$, we write
\( f(T) \lesssim g(T), \) and \( g(T) \gtrsim f(T). \) If both \( f(T) \lesssim g(T) \) and \( f(T) \gtrsim g(T) \), we write \( f(T) \approx g(T). \)

We now begin by establishing a lower bound on the regret any algorithm can achieve, in our setting of the tree-armed bandit problem. Our argument works by reducing to a finite arm space, and then applying a lower bound of Bubeck (2010).

**Theorem 2.** Suppose the trees \( T_i \) have no leaf nodes, and fix \( \beta \geq 0 \). For large enough \( \kappa, \lambda > 0 \), small enough \( \gamma, \varepsilon \in (0, 1) \), and any strategy for tree-armed bandits, we have events \( E_T \) and \( E'_T \), each of probability at least \( \varepsilon \) under some \( P \in \mathcal{P} \), for which

\[
R_T \geq T r_T \text{ on } E_T, \quad S_T \geq r_T \text{ on } E'_T,
\]

for a rate

\[
r_T \gtrsim T^{-1/(\beta+2)}.
\]

This rate matches, up to log factors, the rates in upper bounds which have previously been proved, for example for the zooming algorithm of Kleinberg et al. (2008), or the HOO algorithm of Bubeck et al. (2011b). In the following, we will show that it also matches upper bounds for the adaptive algorithm described in this paper.

We begin by showing that, up to log factors, **Algorithm 1** achieves the same rates, given only knowledge of the quality \( \gamma \). We note that a similar result was stated by Slivkins (2011), in the case of a single finite tree. In the following, we use a novel argument to correct a flaw in the argument of Slivkins,\(^1\) and also extend the result to multiple, infinite trees \( T_i \).

**Theorem 3.** Fix \( \varepsilon, \gamma \in (0, 1) \). Running **Algorithm 1** with error rate \( \varepsilon \) and quality \( \gamma \), for any \( \beta \geq 0 \) and \( \kappa, \lambda > 0 \), we have events \( E_T \), of probability at least \( 1 - \varepsilon \) under any \( P \in \mathcal{P} \), on which

\[
R_T \leq T r_T, \quad S_T \leq r_T,
\]

for a rate

\[
r_T \lesssim (\gamma T)^{-1/(\beta+2)} \times \begin{cases} \log(T), & \beta = 0, \\ \log(T)^{1/(\beta+2)}, & \beta > 0, \end{cases}
\]

uniformly in \( \gamma \).

We have thus shown that **Algorithm 1** achieves good rates of regret, without detailed knowledge of the reward function \( \mu \). Furthermore, the algorithm adapts to the shape of \( \mu \) not only within a single tree \( T_i \), but also by combining the trees in whichever way minimises the zooming dimension \( \beta \).

\(^1\)The proof of Slivkins’ Lemma 4.4(b) incorrectly assumes that all deactivated boxes have been selected.
In the above theorem, Algorithm 1 still required a bound \( \gamma \) on the quality of \( \mu \). As a corollary, however, we can achieve similar rates of regret, up to say an additional log factor, without prior knowledge of \( \mu \).

**Corollary 3.** Fix \( \varepsilon \in (0,1) \). Running Algorithm 1 with error rate \( \varepsilon \) and quality \( \log(T)^{-1} \), for any \( \beta \geq 0, \kappa, \lambda > 0 \) and \( \gamma \in (0,1) \), we have events \( E_T \), of probability at least \( 1 - \varepsilon \) under any \( P \in \mathcal{P} \), on which

\[
R_T \leq T r_T, \quad S_T \leq r_T,
\]

for a rate

\[
r_T \lesssim T^{-1/(\beta+2)} \times \begin{cases} \log(T)^{3/2}, & \beta = 0, \\ \log(T)^{2/(\beta+2)}, & \beta > 0. \end{cases}
\]

We note that in the above construction, Algorithm 1 is no longer an anytime algorithm, as its quality parameter depends on the time horizon \( T \). If an anytime algorithm is desired, one can be constructed using the doubling trick, as in Slivkins (2011); however, we need not consider this further here.

We have thus shown that Algorithm 1 can achieve near-optimal rates of regret, for the optimal combination of trees \( T_i \), without prior knowledge of \( \mu \). We note that, together with Theorem 1, we can use this result to deduce the first part of Corollary 2, our result establishing good rates of regret in continuum-armed bandits.

### 3.5 Efficient implementation

It remains to discuss the implementation of our algorithm. We will show that, for a careful implementation, it can run in almost linear time.

To begin, we will assume that we can efficiently sample from the distributions \( \pi \mid B, B \in \mathcal{B} \). In the case of continuum-armed bandits, these are just uniform distributions over a product of intervals, which are easily sampled from; if the distribution \( \pi \) is less simple, we note that it can always be well approximated by random descent of the trees \( T_i \).

We now discuss how to implement Algorithm 1 efficiently. The key idea is to store the active boxes \( B \) in a priority queue, sorted by their index \( I_t(B) \). The central operation in the algorithm, choosing an active box \( B \) maximising \( I_t \), can then be performed in constant time.

The remaining work lies in efficiently maintaining the set \( \mathcal{A}_t \) of active boxes, and their indices \( I_t \). We note that for active boxes \( B \), the index \( I_t(B) \), width estimate \( W_t(B) \), and confidence radius \( r_t(B) \) are changed only when we choose an arm \( x_t \in B \). We thus need ensure only that these quantities can be updated efficiently when given a new data point.

To do so, we will keep some preliminary computations stored in memory. For each active box \( B \in \mathcal{A}_t \), we store a list of the past data points \( (x_s, Y_s) \), \( s \leq t \), for which \( x_s \in B \). For each box \( C \subseteq B \) satisfying \( \pi(C \mid B) \geq \gamma \),
we further store the number of hits \( n_t(C) \), and average reward \( \mu_t(C) \). After choosing an arm \( x_t \in B \), we update these stored quantities to account for the new data point, and recompute the dependent quantities \( I_t(B) \), \( W_t(B) \) and \( r_t(B) \).

In the event that we change the active set \( \mathcal{A}_t \), any newly-stored quantities can be computed directly from the past data points \( (x_s, Y_s) \), \( s \leq t \). With this procedure, we can then show that our algorithm runs in almost linear time.

**Theorem 4.** On the event \( E_T \), the computational complexity of Algorithm 1 is:

(i) \( O(T \log(T)) \), in the setting of Theorem 3; and

(ii) \( O(T \log(T)^2 + \log^2(p)) \), in the setting of Corollary 3.

Finally, we note that together with Theorem 1, we can then deduce the second part of Corollary 2, our result establishing computational efficiency in continuum-armed bandits.

## 4 Proofs

In this section, we will give proofs of our results. In Section 4.1, we will prove results on zooming continuity; in Section 4.2, prove our lower bounds on the regret; in Section 4.3, give regret bounds for our algorithm; and in Section 4.4, bound their computational cost.

### 4.1 Proofs on zooming continuity

We first establish that the functions in Examples 1 and 2 are zooming continuous. For Example 1, we may argue directly.

**Proof of Example 1.** We first note that for functions (i), the result is a special-case of Example 2. For functions (ii)–(iv), we must establish that these functions do not satisfy the conditions of zooming continuity given by Definition 1.

For each function \( f \), we will give explicit choices of \( U, x, y \) and \( z \), for which part (ii) of that definition does not hold. Let \( \varepsilon \in (0, 1) \), and set \( U = [-\varepsilon^2, \varepsilon - \varepsilon^2] \). Then \( U \) has diameter \( \varepsilon \), and is a neighbourhood in \([0, 1]\) of 0, a global maximum of \( f \). We now consider each function in turn.

(ii) We have

\[
\sup_{x,y \in U: \|x-y\|_U \leq \varepsilon} \frac{|f(x) - f(y)|}{|f(0) - f(\varepsilon^2)|} \geq \frac{|f(0) - f(\varepsilon^2)|}{|f(0) - f(\varepsilon - \varepsilon^2)|} = \frac{1}{1/\log(\varepsilon^4/2)} \to 1/2,
\]

24
as $\varepsilon \to 0$.

(iii) We have

\[
\frac{\sup_{x,y \in U : \|x-y\| \leq \varepsilon} |f(x) - f(y)|}{\sup_{z \in U} |f(0) - f(z)|} \geq \frac{|f(\varepsilon - 2\varepsilon^2) - f(\varepsilon + \varepsilon^2)|}{|f(0) - f(\varepsilon - \varepsilon^2)|} = \frac{-\exp(-\varepsilon^2) + \exp(-\varepsilon^2)}{\exp(-\varepsilon^2)} = 1 - \exp(-\varepsilon^2((1 - 2\varepsilon^{-2} - (1 - \varepsilon)^{-2})) \to 1,
\]
as $\varepsilon \to 0$.

(iv) We have

\[
\frac{\sup_{x,y \in U : \|x-y\| \leq \varepsilon} |f(x) - f(y)|}{\sup_{z \in U} |f(0) - f(z)|} \geq \frac{|f(0) - f(-\varepsilon^2)|}{|f(0) - f(-\varepsilon^2)|} = 1. \quad \Box
\]

For Example 2, we will argue using two lemmas, which allow us to build more complex zooming continuous functions. Our first lemma establishes that elliptical and separable maxima are zooming continuous.

**Lemma 1.** Let $X \subset \mathbb{R}^p$ be a compact product of intervals, and $x^* \in X$. The following functions $f : X \to \mathbb{R}$ are zooming continuous.

(i) $f(x) = -\|A(x - x^*)\|^\alpha$, for a positive-definite matrix $A \in \mathbb{R}^{p \times p}$, and $\alpha > 0$.

(ii) $f(x) = -\sum_{i=1}^p c_i |x_i - x_i^*|^\alpha_i$, for $c_i, \alpha_i > 0$.

**Proof.** We must show that the functions $f$ satisfy the conditions of zooming continuity given by Definition 1. Part (i) of that definition is satisfied as the functions $f$ are continuous, with a unique global maximum at $x^*$. For part (ii) of the definition, we will consider the functions (i) and (ii) separately.

(i) As $A$ is positive definite, it is diagonalisable, with smallest and largest eigenvalues $\lambda$ and $\overline{\lambda}$, $0 < \lambda \leq \overline{\lambda}$. Let $\varepsilon > 0$, and $U$ be any neighborhood in $X$ of $x^*$, having diameter $d \in (0, \varepsilon]$. Further let $x, y \in U$ satisfy $\|x - y\|_U \leq \varepsilon$.

We thus have $\|x - x^*\|, \|y - y^*\| \leq d$, and since $\text{diam}(U) \leq d$, also $\|x - y\| \leq d\varepsilon$. Setting $u = \|A(x - x^*)\|$, $v = \|A(y - x^*)\|$, we obtain

\[
0 \leq u, v \leq \overline{\lambda}d,
\]
and

\[
|u - v| \leq \|A(x - y)\| \leq \overline{\lambda}d\varepsilon.
\]

25
We deduce
\[ |f(x) - f(y)| = |u^\alpha - v^\alpha| \leq (\lambda d)^\alpha \times \begin{cases} \varepsilon^\alpha, & \alpha \leq 1, \\ \alpha \varepsilon, & \alpha \geq 1. \end{cases} \]

Since \( U \) must contain a point \( z \) satisfying \( \|z - x^*\| \geq \frac{1}{2}d \), we also have
\[ \sup_{z \in U} |f(x^*) - f(z)| \geq \left( \frac{1}{2}\lambda d \right)^\alpha. \]

We conclude that
\[ \sup_{U: \text{diam}(U) \leq \varepsilon} \frac{\sup_{x,y \in U: \|x-y\| \leq \varepsilon} |f(x) - f(y)|}{\sup_{z \in U} |f(x^*) - f(z)|} \rightarrow 0, \]
as \( \varepsilon \rightarrow 0 \).

(ii) Let \( \varepsilon > 0 \), and \( U \) be any neighbourhood in \( X \) of \( x^* \), having diameter at most \( \varepsilon \). Also let \( d_i = \text{diam}_i(U) \in (0, \varepsilon] \), and \( x, y \in U \) satisfy \( \|x - y\|_U \leq \varepsilon \).

For \( i = 1, \ldots, p \), set \( u_i = |x_i - x_i^*|, v_i = |y_i - y_i^*| \). We then have
\[ 0 \leq u_i, v_i \leq d_i, \]
and
\[ |u_i - v_i| \leq |x_i - y_i| \leq d_i \varepsilon. \]

We deduce that
\[ |f(x) - f(y)| \leq \sum_{i=1}^{p} c_i |u_i^{\alpha_i} - v_i^{\alpha_i}| \leq \sum_{i=1}^{p} c_i d_i^{\alpha_i} \times \begin{cases} \varepsilon^{\alpha_i}, & \alpha_i \leq 1, \\ \alpha_i \varepsilon, & \alpha_i \geq 1. \end{cases} \]

Since \( U \) must contain points \( z_i \) satisfying \( |z_i - x_i^*| \geq \frac{1}{2}d_i \), we also have
\[ \sup_{z \in U} |f(x^*) - f(z)| \geq \max_{i=1}^{p} c_i \left( \frac{1}{2}d_i \right)^{\alpha_i}. \]

We conclude that
\[ \sup_{U: \text{diam}(U) \leq \varepsilon} \frac{\sup_{x,y \in U: \|x-y\| \leq \varepsilon} |f(x) - f(y)|}{\sup_{z \in U} |f(x^*) - f(z)|} \rightarrow 0, \]
as \( \varepsilon \rightarrow 0 \).

We have thus shown that both functions \( f \) are zooming continuous. \( \square \)

Our second lemma shows that a function with zooming continuous global maxima is itself zooming continuous.
Lemma 2. Let $X \subset \mathbb{R}^p$ be a compact product of intervals, and $f : X \to \mathbb{R}$ be continuous, with finitely many global maxima $x_1^*, \ldots, x_L^*$. For each maximum $x_i^*$, let the function $g_i : X \to \mathbb{R}$ be zooming continuous, with a global maximum at $x_i^*$. If further

$$f(x_i^*) - f(x) = (g_i(x_i^*) - g_i(x))(1 + o(1)),$$

as $x \to x_i^*$, then $f$ is zooming continuous.

Proof. We must show that $f$ satisfies the zooming continuity conditions of Definition 1. Part (i) of that definition is satisfied as $f$ is continuous, with finitely many global maxima.

For part (ii), let $x_i^*$ be a global maximum of $f$, and $g_i$ the associated zooming continuous function. Given $\varepsilon > 0$, let $U$ be a neighbourhood in $X$ of $x_i^*$, having diameter at most $\varepsilon$, and let $x, y \in U$ satisfy $\|x - y\|_U \leq \varepsilon$.

We then have

$$|f(x) - f(y)| = |f(x) - f(x_i^*) + f(x_i^*) - f(y)|$$

$$= |(g_i(x) - g_i(x_i^*)) (1 + o(1)) + (g_i(x_i^*) - g_i(y)) (1 + o(1))|$$

$$= |g_i(x) - g_i(y)| + \sup_{z \in U} |g_i(x_i^*) - g_i(z)|o(1),$$

and

$$\sup_{z \in U} |f(x_i^*) - f(z)| = \sup_{z \in U} |g_i(x_i^*) - g_i(z)|(1 + o(1)),$$

as $\varepsilon \to 0$.

We deduce that

$$\sup_{U : \text{diam}(U) \leq \varepsilon} \sup_{x,y \in U : \|x - y\|_U \leq \varepsilon} \frac{|f(x) - f(y)|}{f(x_i^*) - f(z)}$$

$$= \sup_{U : \text{diam}(U) \leq \varepsilon} \frac{\sup_{x,y \in U : \|x - y\|_U \leq \varepsilon} |g_i(x) - g_i(y)|}{\sup_{z \in U} |g_i(x_i^*) - g_i(z)|} + o(1)$$

$$\to 0,$$

as $\varepsilon \to 0$, since $g_i$ is zooming continuous. As there are finitely many global maxima $x_i^*$, this limit holds uniformly over the choice of maximum. We conclude that $f$ is zooming continuous.

Combining these lemmas, we have that the functions described in Example 2 are zooming continuous. We next establish that zooming continuity implies our grid and quality conditions.

Proof of Theorem 1. We must show that $\mu$ satisfies the conditions of Definition 2; we first establish that $\mu$ is $\mathcal{E}$-measurable. Since $\mathcal{E}$ is the Borel $\sigma$-algebra on $[0, 1]^p$, it is sufficient to note that $\mu$ is continuous.
We now define partitions $B_m$ of $[0, 1]^p$, composed of products of dyadic intervals, and covers $C_m \subseteq B_m$, which satisfy the conditions of Definition 2. Our constructions will depend on a constant $J \in \mathbb{N}$, to be defined; we will show that for $J$ large enough, the conditions of Definition 2 hold.

To begin, let $\mu$ have global maxima $x^*_l$, $l = 1, \ldots, L$, for some $L \in \mathbb{N}$. Define $X_{l, m}$ to be subset of the level set $X_l$ containing those points $x$ which are closer to $x^*_l$ than any other $x^*_p$. As $\mu$ is continuous, and $[0, 1]^p$ compact, we must have $\text{diam}(X_{l, m}) \to 0$ as $\delta \to 0$.

We therefore have some $m_0 \in \mathbb{N}$ after which the level sets $X_{l, m}$ are far apart: for $m \geq m_0$, we have

$$\max_{l=1}^L \text{diam}(X_{l, m}) \leq \frac{1}{4} \min_{l,l'=1}^L \max_{i=1}^p |x^*_i - x^*_{l', i}|. \tag{11}$$

As a consequence, we also have $\bigcup_{l=1}^L X_{l, m} = X_{\delta m}$. We will define our sets $B_m$ and $C_m$ in two ways, depending on whether or not $m \geq m_0$.

For $m < m_0$, we will partition $X$ into a fixed grid of $2^{2p}$ equal boxes. We set $B_m = C_m = G_{1,m}$, where $G_{1,m}$ is the grid given by all products of dyadic intervals of length $2^{-J}$.

For $m \geq m_0$, we will partition $X$ using grids $G_{l,m}$ of approximately $2^{2p}$ equal boxes, each placed over a level set $X_{l, m}$. The covers $C_m$ and partitions $B_m$ will be defined in terms of these grids, so as to satisfy Definition 2.

We define these sets as follows. For $i = 1, \ldots, p$, and $l = 1, \ldots, L$, choose $j_{i,l,m} \in \mathbb{N}$ as large as possible, such that $2^{J-j_{i,l,m}} \geq \text{diam}_i(X_{l, m})$. Then let $G_{l,m}$ be the smallest grid, composed of boxes $B$ satisfying $\text{diam}_i(B) = 2^{-j_{i,l,m}}$, which covers $X_{l, m}$. Further let $C_m$ be the smallest subset of $\bigcup_{l=1}^L G_{l,m}$ which covers $X_{\delta m}$.

We thus have that $G_{l,m}$ is composed of at most $(2^J + 1)^p$ boxes, and for each $l$ and $m$,

$$\text{diam}_i \left( \bigcup G_{l,m} \right) < 2 \text{diam}_i(X_{l, m}) \leq 2 \text{diam}(X_{l, m}). \tag{12}$$

Applying (11), we conclude that for fixed $m \geq m_0$, the grids $G_{l,m}$ cover disconnected regions of $[0, 1]^p$, and hence the boxes $B \in C_m$ are disjoint.

For $m \geq m_0$, we have thus defined the sets $C_m$; it remains to construct the partitions $B_m$. We will construct each $B_m$ inductively, so that $C_m \subseteq B_m$, and $B_m$ is a refinement of $B_{m-1}$.

Since $j_{i,l,m}$ is non-decreasing in $m$, and $j_{i,l,m} \geq J$, every box $B \in C_m$ must lie within a box $C \in C_{m-1} \subseteq B_{m-1}$. If we first set $B_m = B_{m-1}$, and then subdivide boxes $B \in B_m$ as necessary to ensure $C_m \subseteq B_m$, we will therefore obtain the desired $B_m$.

We have thus constructed the partitions $B_m$ of $[0, 1]^p$, and covers $C_m \subseteq B_m$. It remains to show that, for a large enough choice of $J$, these sets satisfy conditions (i)–(vi) of Definition 2; we consider each condition in turn.
(i) For \( m < m_0 \), \( C_m \) covers \([0,1]^p\), so certainly covers \( X_{\delta_m} \). For \( m \geq m_0 \), \( C_m \) covers \( \bigcup_{l=1}^L X_{l,\delta_m} = X_{\delta_m} \).

(ii) For \( m < m_0 \), \( C_m \) has cardinality \( 2^J p \); for \( m \geq m_0 \), it has cardinality at most \( L(2^J + 1)p \). In both cases, its cardinality is thus bounded by \( \kappa \delta_m^\beta \), for constants \( \kappa > 0 \) and \( \beta = 0 \).

(iii) We first prove part (iiiA), and then deduce part (iiiB).

(a) First suppose \( m \geq m_0 \), and set \( C_{l,m} = C_m \cap G_{l,m}, U_{l,m} = \bigcup C_{l,m} \).

Then \( U_{l,m} \) is a neighbourhood in \( X \) of \( x^*_l \), and from (12), we have \( \text{diam}(U_{l,m}) \to 0 \) as \( m \to \infty \).

We will use the fact that \( \mu \) is zooming continuous, applied to \( U_{l,m} \).

Set \( \delta_{l,m} = \sup_{z \in U_{l,m}} |\mu(x^*_l) - \mu(z)| \). We have that for \( B \in C_{l,m} \),

\[
W(B) \leq \frac{\sup_{x,y \in U_{l,m}} \|x-y\|_{U_{l,m}} \sqrt{p^{2^{-J}} - 1} |\mu(x) - \mu(y)|}{\delta_{l,m}} \leq \frac{1}{13p},
\]

for large enough \( J \), and \( m \geq m_1 \), for some \( m_1 \geq m_0 \).

Now pick \( B \in C_{l,m} \) for which \( \inf_{z \in B} \mu(z) = \inf_{z \in U_{l,m}} \mu(z) \). By the definition of \( C_{l,m} \), we must have some \( x \in B \) for which \( \mu(x) \geq \mu(x^*_l) - \delta_m \). Using (13), we then have

\[
\delta_{l,m} = \sup_{z \in B} |\mu(x^*_l) - \mu(z)| = \sup_{z \in B} (\mu(x^*_l) - \mu(x) + \mu(x) - \mu(z)) \leq \delta_m + \delta_{l,m}/13p.
\]

Since \( p \geq 1 \), we conclude that

\[
\delta_{l,m} \leq 13\delta_m/12, \quad (14)
\]

and thus for \( B \in C_{l,m} \),

\[
W(B) \leq \delta_m/12p.
\]

Choosing \( J \) and \( m_1 \) large enough, this holds for all \( B \in C_m \), and \( m \geq m_1 \).

For \( m < m_1 \), we note that since \( \mu \) is continuous on a compact space, it is uniformly continuous. We therefore have that for \( J \) large enough,

\[
W(B) \leq \delta_m/12p \leq \delta_m/12p,
\]

for all \( B \in C_m, m < m_1 \).
(b) For each \( l = 1, \ldots, L \), and \( m \geq m_0 \), we have some \( B \in C_{l,m} \) containing \( x_l^* \). By part (iii), we then have

\[
\delta(B) = W(B) \leq \delta_{m}/12p \leq \delta_{m+1},
\]

so \( B \subseteq X_{l,\delta_{m+1}} \), and \( j_{i,l,m+1} \leq j_{i,l,m} + J \). Inductively, we conclude that for all \( B \in C_m, m \in \mathbb{N} \), we have \( d(B) \leq \lambda m \), for a constant \( \lambda > 0 \).

(iv) Let \( B \) be a box on \( C_m, m \in \mathbb{N} \). If \( W(B) \leq \frac{1}{4} \delta(B) \), the required result trivially holds; we will assume not. Suppose first that \( m \geq m_0 \).

As the grids \( G_{l,m} \) cover disconnected regions, we must have \( B \) is on some \( C_{l,m} \).

Let \( U_B = X_{l,\delta(B)} \); from (14), we have that \( \delta(B) \to 0 \) as \( m \to \infty \), so \( \text{diam}(U_B) \to 0 \). Given also \( K \in \mathbb{N} \), we can subdivide \( B \) into a grid \( G_B \) of \( 2^K \) boxes, whose dimensions are \( 2^{-K} \)-times those of \( B \).

Applying the zooming continuity of \( \mu \) to the neighbourhood \( U_B \) in \( X \) of \( x_l^* \), we then obtain for any \( C \in G_B \),

\[
W(C) \leq \frac{1}{32} \delta(B) \leq \frac{1}{8} W(B).
\]

Now take two boxes \( C_0, C_p \in G_B \), satisfying

\[
\inf_{z \in C_0} \mu(z) = \inf_{z \in B} \mu(z), \quad \sup_{z \in C_p} \mu(z) = \sup_{z \in B} \mu(z).
\]

From (15), we may conclude that

\[
\mu(C_p) - \mu(C_0) \geq \inf_{z \in C_p} \mu(z) - \sup_{z \in C_0} \mu(z) \geq \frac{3}{4} W(B).
\]

If we further choose a sequence \( C_0, \ldots, C_p \) of boxes in \( G_B \), with the property that each \( C_{i+1} \) agrees with \( C_i \) except along one axis, then we must have some \( C_1, C_{i+1} \) which satisfy

\[
\mu(C_{i+1}) - \mu(C_i) \geq \frac{3}{4p} W(B) \geq \frac{1}{p}(W(B) - \frac{1}{4}\delta(B)).
\]

30
As $K$ can be chosen to be a fixed constant, we also have that
\[
\pi(C_i \mid B), \pi(C_{i+1} \mid B) \geq \gamma,
\]
for the fixed constant $\gamma = 2^{-pK}$.

(v) Each $C_m$ is a subset of the finite union the of grids $G_l,m$. By construction, for each $m \in \mathbb{N}$, the grids $G_l,m$ cover disconnected regions, so are separated.

(vi) Each $B_{m+1}$ is a refinement of $B_m$, by construction.

We hence conclude that $\mu$ satisfies Definition 2, with zooming dimension $\beta = 0$. \hfill \square

4.2 Proofs of lower bounds

We now prove our lower bounds on the regret. To begin, we will need a lower bound on the regret over a finite arm space, as proved by Bubeck (2010).

**Lemma 3.** Let $\frac{1}{3} \leq u < v \leq \frac{2}{3}$, and set $\Delta = v - u$. Consider a multi-armed bandit problem, with arms $X = \{1, \ldots, K\}$. Let the reward distributions $P(x)$ be Bernoulli, with mean function
\[
\mu_k(x) = \begin{cases} 
  v, & x = k, \\
  u, & x \neq k,
\end{cases}
\]
for some $k \in X$. Further let $P_k$ and $E_k$ denote probability and expectation under the reward function $\mu_k$. Then for any strategy, we have
\[
\min \left( \frac{1}{K} \sum_{k=1}^{K} E_k \frac{R_T}{T\Delta}, \frac{1}{K} \sum_{k=1}^{K} E_k \frac{S_T}{\Delta} \right) \geq 1 - \frac{1}{K} - \frac{3}{2} \sqrt{\frac{T\Delta^2}{K}}. \tag{16}
\]

**Proof.** The result is proved similarly to Lemma 2.2 of Bubeck (2010). We begin with the cumulative regret $R_T$, and note that
\[
\frac{1}{K} \sum_{k=1}^{K} E_k \frac{R_T}{T\Delta} \geq 1 - \frac{1}{K} \sum_{k=1}^{K} E_k \frac{R_T}{T\Delta}
\]
\[
= 1 - \frac{1}{KT} \sum_{t=1}^{T} \sum_{k=1}^{K} P_k(x_t = k).
\]

Arguing as in Bubeck (2010), we may assume our strategy is deterministic, and then for $t \leq T$ we have
\[
\frac{1}{K} \sum_{k=1}^{K} P_k(x_t = k) \leq \frac{1}{K} + \frac{1}{K} \sum_{k=1}^{K} \sqrt{\frac{1}{2} KL(P_0, P_k)}
\]
\[
\leq \frac{1}{K} + \sqrt{\frac{T}{2K} KL(u, v)},
\]
31
where \( P_0 \) denotes probability under the reward function \( \mu(x) = u \), and \( KL(u, v) \) denotes the Kullback-Leibler divergence of Bernoulli random variables with means \( u \) and \( v \). As this divergence is bounded by the \( \chi^2 \) divergence, we further have

\[
KL(u, v) \leq \frac{(v - u)^2}{u(1 - u)} \leq \frac{9}{2} \Delta^2.
\]

Combining the above inequalities gives the result for \( R_T \). For \( S_T \), we likewise have that

\[
\max_{k=1}^K \mathbb{E}_k \frac{S_T}{\Delta} \geq 1 - \frac{1}{K} \sum_{k=1}^K P_k(\hat{x}_T = k),
\]

which can be bounded in a similar fashion. \( \square \)

Next, we prove that we can apply this lemma in a more general bandit setting.

**Lemma 4.** Let \( \frac{1}{3} \leq u < v \leq \frac{2}{3} \), and set \( \Delta = v - u \). Consider a multi-armed bandit problem whose arm space \( X \) can be partitioned into sets \( \Omega_0, \ldots, \Omega_K \). Let the reward distributions \( P(x) \) be Bernoulli, with reward function

\[
\mu_k(x) = \begin{cases} 
v, & x \in \Omega_k, \\ u, & x \in \Omega_j, j = 1, \ldots, K, j \neq k, \\ \mu(x), & x \in \Omega_0, 
\end{cases}
\]

for some \( k = 1, \ldots, K \), and a function \( \mu(x) \leq u \). Further let \( P_k \) and \( E_k \) denote probability and expectation under the reward function \( \mu_k \). If \( K \geq 2 \), and

\[
\sqrt{\frac{T \Delta^2}{K}} \leq \frac{1}{6}, \tag{17}
\]

then for any strategy, we have

\[
\min \left( \max_{k=1}^K P_k \left( \frac{R_T}{T \Delta} \geq \frac{1}{8} \right), \max_{k=1}^K P_k \left( \frac{S_T}{\Delta} \geq \frac{1}{8} \right) \right) \geq \frac{1}{8}. \tag{18}
\]

**Proof.** We will reduce the multi-armed bandit problem (A) in Lemma 4 to the multi-armed bandit problem (B) in Lemma 3: given a strategy \( \hat{x}_t \) for (A), we will construct a strategy \( \hat{x}'_t \) for (B) whose regret is no larger. The result will then follow from Lemma 3.

We begin with the cumulative regret \( R_T \). Let \( Y' \) denote the rewards in (B), and \( \hat{x}_t \) denote any strategy for (A). Generate independent random variables \( Z'_t = (Z_t, V_t) \), where \( Z_t \) is distributed as in the randomisation of \( \hat{x}_t \), and
$V_t$ is independently uniform on $[0, 1]$. Let $x_t = x_t(Y_1, \ldots, Y_{t-1}, Z_1, \ldots, Z_t)$, where

$$Y_t = \begin{cases} Y'_t, & x_t \notin \Omega_0, \\ 1(V_t \leq \mu(x_t)), & x_t \in \Omega_0. \end{cases}$$

Then define a strategy $x'_t$ for (B) by

$$x'_t(Y'_1, \ldots, Y'_{t-1}, Z'_1, \ldots, Z'_t) = \begin{cases} j, & x_t \in \Omega_j, j = 1, \ldots, K, \\ 1, & x_t \in \Omega_0. \end{cases}$$

If we apply the strategy $x'_t$ in (B), then the $Y_t$ and $Z_t$ are distributed exactly as if the strategy $x_t$ were applied in (A). We thus have that

$$R_T = \sum_{t=1}^{T} (v - \mu_k(x_t))$$

is distributed as the regret in (A) under $x_t$. Letting $\mu'_k$ denote the reward function of (B), we also have $\mu'_k(x'_t) \geq \mu_k(x_t)$, so the regret in (B) under $x'_t$,

$$R'_T = \sum_{t=1}^{T} (v - \mu'_k(x'_t)),$$

satisfies

$$R'_T \leq R_T. \quad (19)$$

We may thus bound $R_T$ using Lemma 3. From (16) and (17), we have that

$$\max_{k=1}^{K} \mathbb{E}_k \frac{R'_T}{T\Delta} \geq \frac{1}{4}.$$

Since $R'_T/T\Delta \in [0, 1]$, we deduce that

$$\max_{k=1}^{K} \mathbb{P}_k \left( \frac{R'_T}{T\Delta} \geq \frac{1}{8} \right) \geq \frac{1}{8}.$$

The desired result then follows from (19).

For the simple regret $S_T$, suppose we are also given an maximum estimate $\hat{x}_T$ for (A). We may then likewise generate a random variable $\hat{Z}_T$ as in the randomisation of $\hat{x}_T$; let

$$\hat{x}_T = \hat{x}_T(Y_1, \ldots, Y_T, Z_1, \ldots, Z_1, \hat{Z}_T);$$

and define an estimated maximum $\hat{x}'_T$ in (B) by

$$\hat{x}'_T(Y'_1, \ldots, Y'_T, Z'_1, \ldots, Z'_T, \hat{Z}_T) = \begin{cases} j, & \hat{x}_T \in \Omega_j, j = 1, \ldots, K, \\ 1, & \hat{x}_T \in \Omega_0. \end{cases}$$

Letting $S_T$ denote the simple regret of $\hat{x}_T$ in (A), and $S'_T$ the simple regret of $\hat{x}'_T$ in (B), we have $S'_T \leq S_T$, so may argue as before.
We may now prove our tree-armed bandit lower bound.

Proof of Theorem 2. We will proceed by a reduction to a multi-armed bandit problem. For each $T \in \mathbb{N}$, we will construct specific arm distributions $P_k \in \mathcal{P}$, $k = 1, \ldots, K$, to which we can apply Lemma 4.

To proceed, we will first assume $\beta > 0$; we return later to the case $\beta = 0$. We will define a class of reward distributions $P_{l,k}$, for $l \in \mathbb{N}$, and $k = 1, \ldots, 2^l$; the specific $P_k$ we consider in Lemma 4 will then be given by the $P_{l,k}$ for a fixed $l$, chosen in terms of $T$.

We will define the reward distributions $P_{l,k}$ to be Bernoulli, each having some reward function $\mu_{l,k}$. The functions $\mu_{l,k}$ will be defined so that for fixed $l$, these functions agree except in the location of their global maximum. The parameter $k$ will then index the $2^l$ possible choices for that location.

In order for the distributions $P_{l,k}$ to lie within $\mathcal{P}$, we will require the functions $\mu_{l,k}$ to have many local maxima, centred around the possible locations of their global maximum. We will thus need to carefully construct the $\mu_{l,k}$ in terms of the trees $T_i$.

We therefore define a collection of nodes $U_{i,j} \in T_1$, as follows. Set $U_{0,1} = X_1$, and for each $i = 0, 1, \ldots$ and $j = 1, \ldots, 2^i$, given $U_{i,j}$, pick two children $U_{i,j}^1, U_{i,j}^2$ of $U_{i,j}$ in $T_1$. Then set $U_{i+1,2j+1}^k$ to be a child of $U_{i,j}^k$, for $k = 1, 2$. For $l \in \mathbb{N}$, $k = 1, \ldots, 2^l$, we may now define the reward functions

$$\mu_{l,k}(x) = \frac{1}{3} \left( 1 + (1 - \alpha) \sum_{i=0}^{l-1} \alpha^i 1 \left( x_1 \in \bigcup_{j=1}^{2^i} U_{i,j} \right) + \alpha^l 1 \left( x_1 \in U_{l,k} \right) \right),$$

where the constant $\alpha = 2^{-1/\beta}$.

We note that $\mu_{l,k}$ has many local maxima, on sets $U_{l-1,j} \times X_2 \times \cdots \times X_p$, $j = 1, \ldots, 2^{l-1}$, but is globally maximised only on $U_{l,k} \times X_2 \times \cdots \times X_p$. Since $X_1$ has at least four grandchildren in $T_1$, we have

$$\inf_{z \in X} \mu_{l,k}(z) = \frac{1}{3} (2 - \alpha) \geq \frac{1}{3},$$

likewise, we also have

$$\sup_{z \in X} \mu_{l,k}(z) = \frac{2}{3}.$$

To proceed, we must first show that the reward functions $\mu_{l,k}$ satisfy Definition 2. We note that since the $\mu_{l,k}$ are $\mathcal{E}$-simple functions, they are certainly $\mathcal{E}$-measurable. We must now show that, for fixed $l \in \mathbb{N}$, $k = 1, \ldots, 2^l$, and reward function $\mu = \mu_{l,k}$, we can construct partitions $B_m$ of $X$, and covers $C_m \subseteq B_m$, satisfying the conditions of Definition 2.

To begin, for each $m \in \mathbb{N}$, we define a parameter $i_m \in \mathbb{N}_0$ as follows. If $\delta_m \geq \frac{1}{3}$, we set $i_m = 0$. If $\delta_m < \frac{1}{3} \alpha^l$, we set $i_m = l$. Otherwise, we choose $i_m \in \mathbb{N}_0$ as large as possible, subject to $\delta_m < \frac{1}{3} \alpha^i m$. 34
We also fix a constant $I \in \mathbb{N}$ such that $\alpha^I \leq 1/4p$. We then set $\mathcal{B}_m$ to be the collection of boxes $U \times X_2 \times \cdots \times X_p$, for all nodes $U$ of depth $2(i_m + I)$ in $\mathcal{T}_1$. We further set $\mathcal{C}_m$ to be the subset of such boxes for which $U \subseteq \bigcup_{j=1}^{2^m} U_{i_m,j}$.

We note that $\mathcal{B}_m$ is indeed a partition of $X$, and $\mathcal{C}_m \subseteq \mathcal{B}_m$. It remains to check that these sets satisfy conditions (i)-(vi) of Definition 2. We consider each condition in turn.

(i) We have $X_{\delta_m} \subseteq \bigcup_{j=1}^{2^m} U_{i_m,j} \times X_2 \times \cdots \times X_p$, so $\mathcal{C}_m$ indeed covers $X_{\delta_m}$.

(ii) The cardinality of $\mathcal{C}_m$ is at most $2^{mq}e^{2I}$, which is bounded by $\kappa \delta_m^{-\beta}$ for large enough $\kappa > 0$.

(iii) For each box $B \in \mathcal{C}_m$, we have the following.

(a) If $i_m = l$, then $\mu_{l,k}$ is constant on $B$, so $W(B) = 0$. Otherwise, either $\mu_{l,k}$ is constant on $B$, or $\inf_{x \in B} \mu_{l,k}(x) \geq \frac{1}{3}(2 - \alpha^{i_m+I+1})$, so we have $W(B) \leq \frac{1}{3}\alpha^{i_m+I+1} \leq \delta_m/12p$.

(b) $d(B) = 2(i_m + I) \leq \lambda m$, for a constant $\lambda > 0$.

(iv) Let $B$ be a box on $\mathcal{C}_m$. We must have $B = U \times X_2 \times \cdots \times X_p$, for a node $U \in \mathcal{T}_1$. We consider separately the cases:

(a) $U$ is equal to some $U_{i,j}$;

(b) $U$ is a child of some $U_{i,j}$; and

(c) otherwise.

In case (iva), we set $J = \lceil 2\beta \rceil$, and first assume that $i + J < l$. We then have some nodes $U_1, U_2$, of depth at most $2(i + J)$ in $\mathcal{T}_1$, for which the boxes $C_k = U_k \times X_2 \times \cdots \times X_p$ satisfy $C_1, C_2 \subseteq B$, and $\mu_{l,k}(x) \geq \frac{1}{4}(2 - \alpha^{i+J+1})$ on $C_1$, $\mu_{l,k}(x) = \frac{1}{4}(2 - \alpha^{i+1})$ on $C_2$.

Since $W(B) \leq \frac{3}{3}\alpha^{i+1}$, we conclude that $\mu_{l,k}(C_1) - \mu_{l,k}(C_2) \geq (1 - \alpha^J)W(B) \geq \frac{3}{4}W(B) \geq \frac{1}{p}(W(B) - \frac{1}{2}\delta(B))$.

If instead $i + J \geq l$, then we can similarly find nodes $U_1, U_2$, of depth at most $2(i + J)$ in $\mathcal{T}_1$, for which the boxes $C_1, C_2 \subseteq B$ satisfy $\mu_{l,k}(C_1) - \mu_{l,k}(C_2) = W(B) \geq \frac{1}{p}(W(B) - \frac{1}{2}\delta(B))$.

In either of the above cases, we also have $\pi(C_k \mid B) \geq \gamma$, $k = 1, 2$,
for the constant $\gamma = q^{-2l'}$.

In case (ivb), if $\mu$ is constant on $B$, the result is trivial; otherwise, the result follows by the same argument as in case (iva). Finally, in case (ivc), $\mu$ must be constant on $B$, so the result is again trivial.

(v) By construction, $C_m$ is itself a single grid $G_{1,m}$.

(vi) By construction, $B_{m+1}$ is a refinement of $B_m$.

We thus conclude that the reward functions $\mu_{l,k}$ satisfy Definition 2. It remains to lower bound the regret over the reward distributions $P_{l,k}$, using Lemma 4.

For $T \in \mathbb{N}$, we choose $l \in \mathbb{N}$ as small as possible, subject to $2^{l(1+2/\beta)} \geq 4T$, and set $K = 2^l$. We then apply Lemma 4 to the reward distributions $P_{l,k}$, $k = 1, \ldots, K$, taking the sets $\Omega_k = U_{1,k} \times X_2 \times \cdots \times X_p$, $k = 1, \ldots, K$, and $\Omega_0 = X \setminus \bigcup_{k=1}^K \Omega_k$. We have that the quantity $\Delta = \frac{1}{2} 2^{-l/\beta}$, so we satisfy condition (17). The desired result then follows from (18), letting $E_T$ be the event that $R_T \geq T\Delta/8$, and $E_T'$ the event that $S_T \geq T/8$.

We have thus proved the theorem in the case $\beta > 0$. For $\beta = 0$, we proceed in similar fashion. For $k = 1, 2$ we define $\mu_{1,k}$ and $P_{1,k}$ as above, now setting $\alpha = 1/\sqrt{2T}$. In this case, Definition 2 is trivially satisfied, setting $B_m = C_m$ to be the collection of boxes $U \times X_2 \times \cdots \times X_p$, for all nodes $U$ of depth 2 in $T_1$.

We then apply Lemma 4 to the reward distributions $P_{1,1}$ and $P_{1,2}$, as above. We have $K = 2$ and $\Delta = 1/3\sqrt{2T}$, so again satisfy condition (17); the result again follows from (18).

4.3 Proofs of regret bounds

We next focus on the regret of Algorithm 1. Our proofs are based on those of Slivkins (2011), although we make some significant changes, both to correct a flaw in Slivkins’ argument, and also to provide improved results.

We begin by describing an event which occurs with high probability, on which we can control the behaviour of our algorithm. We may then argue deterministically, conditional on that event.

**Definition 3.** We will say an execution of Algorithm 1 is clean if, for any $t \in \mathbb{N}$, the following holds.

(i) If $B \in B$ and $n_t(B) > 0$, $|\mu_t(B) - \mu(B)| \leq r_t(B)$.

(ii) For each box $B$ on a cover $C_m$, $m \in \mathbb{N}$, fix two boxes $C_1, C_2 \subseteq B$, satisfying the conditions of Definition 2(iv). If $r_t(B) \leq \sqrt{\gamma/6}$, then $r_t(C_k) \leq \sqrt{2/\gamma}r_t(B)$, for $k = 1, 2$. 36
In other words, an execution of Algorithm 1 is clean if the confidence radii \( r_t \) bound the error in the estimates \( \mu_t \), and we have two boxes \( C_1, C_2 \), satisfying the conditions of Definition 2(iv), which are hit by enough arms \( x_t \). We can show that both these events occur with high probability.

**Lemma 5.** In the setting of Theorem 3, an execution of Algorithm 1 is clean with probability at least \( 1 - \varepsilon \).

**Proof.** We will consider separately the two conditions in Definition 3, and show that the probability of each one failing is at most \( \frac{1}{2} \varepsilon = 2\tau^{-1} \).

(i) Let \( H_t(B) \) denote the event that the box \( B \) was hit at time \( t \), and for any \( B \in \mathcal{B} \) and \( i \in \mathbb{N} \), define the random variables

\[
V_i(B) = \begin{cases} 
Y_t - \mu(B), & n_t(B) = i, H_t(B), \\
0, & n_t(B) < i \text{ for all } t.
\end{cases}
\]

The martingale \( M_n(B) = \sum_{i=1}^{n} V_i(B) \) then satisfies

\[
\mu_t(B) - \mu(B) = M_{n_t(B)}(B)/n_t(B).
\]

If Definition 3(i) does not hold, we must have

\[
|M_n(B)| > 2\sqrt{n \log[p(B)(\tau + n)]},
\]

for some \( B \in \mathcal{B} \) and \( n \in \mathbb{N} \). For a single \( B \) and \( n \), by Azuma’s inequality, the event (20) has probability at most \( 2[p(B)(\tau + n)]^{-2} \).

Now, the number of boxes \( B \) of depth \( d \) is bounded by

\[
\sum_{j=0}^{d} q^j \leq q^{d+1},
\]

so by a union bound, the probability that any event (20) occurs is at most

\[
\sum_{n=1}^{\infty} \sum_{d=0}^{\infty} q^{p(d+1)}2[p^{p(d+1)}(\tau + n)]^{-2} = 2 \sum_{n=\tau+1}^{\infty} n^{-2} \sum_{d=1}^{\infty} q^{-pd} \leq 2\tau^{-1}.
\]

(ii) Given \( B, C_1, \) and \( C_2 \) as in Definition 3(ii), for \( i \in \mathbb{N} \) and \( k = 1, 2 \), we likewise define the random variables

\[
V_{i,k}(B) = \begin{cases} 
1(H_t(C_k)) - \pi(C_k \mid B), & n_t(B) = i, H_t(B), \\
0, & n_t(B) < i \text{ for all } t.
\end{cases}
\]

The martingale \( M_{n,k}(B) = \sum_{i=1}^{n} V_{i,k}(B) \) then satisfies

\[
n_t(C_k) - n_t(B)\pi(C_k \mid B) = M_{n_t(B),k}(B).
\]
If Definition 3(ii) does not hold, we must have \( r_t(C_k) > \sqrt{2/\gamma} r_t(B) \), and so

\[
n_t(C_k) < \frac{1}{2} \gamma n_t(B),
\]

for some \( B, k \) and \( t \). Since \( \pi(C_k | B) \geq \gamma \), we then have

\[
M_{n,k}(B) < -\frac{1}{2} \gamma n_t(C_k | B),
\]

for \( n = n_t(B) \). For Definition 3(ii) to not hold, we must also have \( r_t(B) \leq \sqrt{\gamma/6} \), so we further deduce that

\[
\gamma n \geq 24 \log[\rho(B)(\tau + n)].
\]

For a single such \( B, k \) and \( n \), by Freedman’s inequality, the event (21) has probability at most \( \rho(B)(\tau + n) \). Arguing as in part (i), the probability that any event (21) occurs is thus at most \( 2\tau^{-1} \). \( \square \)

We may now begin to prove our regret bound for Algorithm 1. We will first show that the active boxes \( B \) must lie on the covers \( C_m \); once this is proved, we will be able to precisely control the behaviour of our algorithm.

In the following, we will let \( A_t \) denote the set of boxes active at time \( t \), after ensuring Invariant 1. We will also say a box \( B \) was activated at time \( t \), if it was made active at some point during that time step; we will consider the box \( X \) to have been activated at time 0.

**Lemma 6.** Let \( P(t) \) be the statement that, if \( B \) was activated at a time \( s \in \mathbb{N}, s \leq t, \) and \( 2^{-m} \leq \delta(B_s) \leq 2^{-m} \), then \( B \) is on the cover \( C_m \). Then in the setting of Theorem 3, on a clean execution, for any \( B \in B \) and \( t \in \mathbb{N} \):

(i) if \( P(t) \) holds, and \( B \in A_t \), then \( \frac{4}{3} W(B) - \frac{1}{3} \delta(B) \leq 2p\nu r_t(B) \);

(ii) if \( P(t-1) \) holds, then \( \Delta(B_t) \leq 2(p\nu+1)r_{t-1}(B_t) \);

(iii) if \( P(t-1) \) holds, and \( B \subseteq B_t \), then \( \delta(B_t) \leq 5p\nu r_t(B) \); and

(iv) \( P(t-1) \) holds.

**Proof.** We will first establish parts (i)–(iii), and then use these results to prove part (iv) inductively.

(i) We will show that, since \( B \) satisfies Invariant 1, its width \( W(B) \) cannot be much larger than its confidence radius \( r_t(B) \). To begin, we will need to prove that \( B \) lies on some cover \( C_m \).

If \( B = X \), then \( B \) is on the cover \( C_1 \). Any other \( B \in A_t \) must have been activated at some time \( s \in \mathbb{N}, s \leq t; \) if \( \delta(B_s) > 0 \), then by \( P(t) \), \( B \) must be on some cover \( C_m \). In the remaining case, since \( B \subseteq B_s \), we have

\[
W(B) \leq \delta(B) \leq \delta(B_s) = 0,
\]
and the result is trivial.

We may thus assume $B$ lies on some cover $C_m$, and so let $C_1, C_2 \subseteq B$ be the boxes fixed in Definition 3(ii). If $r_t(B) > \sqrt{\gamma/6}$, then $2p\nu r_t(B) \geq 1$, so the desired result follows from $W(B) \leq \delta(B) \leq 1$. Otherwise, we have

$$
\nu r_t(B) \geq \mu(B) - \delta(B) - 2r_t(B) \geq 2r_t(B) - 2r_t(C_1) - 2r_t(C_2) \\
\geq \frac{3}{4} \left[ \frac{4}{3} W(B) - \frac{1}{4} \delta(B) \right] - \frac{1}{2} \nu r_t(B),
$$

where the first inequality follows from Invariant 1, the second from Definition 3(i), and the third from Definitions 2(iv) and 3(ii). Rearranging gives the desired result.

(ii) We will show that, since the box $B_t$ maximises the index $I_{t-1}$, its mean $\mu(B)$ must be close to the optimum $\mu^*$. First, we pick a box $B_t^* \in \mathcal{A}_{t-1}$ satisfying $\sup_{z \in B_t^*} \mu(z) = \mu^*$. By part (i), we have

$$
W(B_t^*) = \frac{4}{3} W(B_t^*) - \frac{1}{4} \delta(B_t^*) \leq 2p\nu r_{t-1}(B_t^*). \tag{22}
$$

We then deduce that

$$
I_{t-1}(B_t) \geq I_{t-1}(B_t^*) \geq \mu(B_t^*) + W(B_t^*) \geq \mu^*, \tag{23}
$$

where the first inequality follows since $B_t$ maximises $I_{t-1}$, the second from (22) and Definition 3(i), and the third from the definition of $B_t^*$. We conclude that

$$
\Delta(B_t) \leq \mu^* - \mu_{t-1}(B_t) + r_{t-1}(B_t) \leq 2(p\nu + 1)r_{t-1}(B_t),
$$

where the first inequality follows again using Definition 3(i), and the second from (23).

(iii) We will show that we can bound $r_t(B)$ by $r_{t-1}(B_t)$, and then apply parts (i) and (ii). First, we note that since $B \subseteq B_t$,

$$
n_t(B) \leq n_{t-1}(B_t) + 1. \tag{24}
$$

If $n_{t-1}(B_t) \leq 24$, then $n_t(B) \leq 25$, so $5p\nu r_t(B) \geq 1$, and the desired result follows from $\delta(B_t) \leq 1$.

Otherwise, we have

$$
\delta(B_t) \leq \Delta(B_t) + W(B_t) \\
\leq \frac{1}{2} (7p\nu + 4)r_{t-1}(B_t) + \frac{1}{4} \delta(B_t), \tag{25}
$$

39
where the first inequality follows from (10), and the second from parts (i) and (ii). We thus obtain
\[ \delta(B_t) \leq \frac{2}{3}(7p\nu + 4)r_{t-1}(B_t) \leq 5p\nu r_t(B), \]
where the first inequality follows from (25), and the second from (24).

(iv) We will now show the statements \( P(t) \) hold, arguing by induction on \( t \). We first note that the statement \( P(0) \) is trivial; for \( t \in \mathbb{N} \), we now suppose \( P(t-1) \) holds, but \( P(t) \) does not, and show this leads to a contradiction.

Let \( B \) be the box, activated at time \( t \), which is the first-activated box to not satisfy \( P(t) \). Further let \( C \) be the box deactivated immediately before \( B \) was activated. Note that we either have \( C = B_t \), or that \( C \) was activated at time \( t \), so \( C \subset B_t \).

We will need to establish that \( C \) lies on a single grid \( G_{l,m} \), where \( m \in \mathbb{N} \) satisfies
\[ 2^{-m} \leq \delta(B_t) \leq 2^{1-m}. \]
To show this, we first prove that \( C \) lies on \( B_m \).

If \( C = X \), the statement is trivial. Otherwise, \( C \) must have been activated before \( B \), at some time \( s \leq t \). Since \( B \subset B_t \), and \( B \subset C \subset B_s \), we must have \( B_t \subset B_s \), and
\[ \delta(B_s) \geq \delta(B_t) \geq 2^{-m}. \]
By the inductive hypothesis, \( C \) lies on some partition \( B_l \), \( l \leq m \), and then by Definition 2(vi), also on \( B_m \).

We thus have that \( C \) lies on \( B_m \). Since \( C \subset B_t \), we also have
\[ \delta(C) \leq \delta(B_t) \leq 2^{1-m}, \]
so \( C \) further lies on the cover \( C_m \). Using Definition 2(v), \( C \) must then lie on a single grid \( G_{l,m} \).

We will now proceed to show a contradiction. Let \( G_{l,m} \) be generated by subsets \( S_j \) of each tree \( T_j \), and set \( C = \prod_{j=1}^{p} V_j \), \( V_j \in T_j \). As \( C \) was deactivated at time \( t \), we must have boxes \( C_1, C_2 \subset C \), differing only in axis \( i \), for which
\[ \mu_t(C_1) - \mu_t(C_2) \geq \nu r_t(C). \quad (26) \]

We first suppose \( V_i \not\in S_i \), and show this leads to a contradiction. In this case, as \( C \) is on \( G_{l,m} \), and \( B \) was formed by splitting \( C \) along axis \( i \), we must have that \( B \) is on \( G_{l,m} \). Since \( C \) is on \( C_m \), and \( B \subset C \), we conclude that \( B \) is also on \( C_m \), contradicting our choice of \( B \).
We must therefore have $V_i \in S_i$; we will show that this too leads to a contradiction. Suppose we have two points $x_1, x_2 \in C$, which agree except in the $i$-th coordinate. Then these points must lie within the same member of $C_m$, so

$$|\mu(x_1) - \mu(x_2)| \leq \frac{1}{6p}2^{-m}. \quad (27)$$

Now let $U_{i,k}$, $k = 1, 2$, and $U_j$, $j \neq i$, be defined in terms of $C_1, C_2$, as in (9). Further let $\bar{\pi}_0$ denote the product distribution of $\pi_j \mid U_j$, $j \neq i$, let $\bar{\pi}_k$ denote the distribution $\pi_i \mid U_{i,k}$, and for $x \in X$, let $x_{-i}$ denote the vector $(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_p)$.

We then obtain

$$\mu_t(C_1) - \mu_t(C_2) \leq \mu(C_1) - \mu(C_2)$$

$$= \int \int \mu(x) d(\bar{\pi}_1 - \bar{\pi}_2)(x_i) d\bar{\pi}_0(x_{-i})$$

$$\leq \int \frac{1}{6p}2^{-m} d\bar{\pi}_0(x_{-i})$$

$$= \frac{1}{6p}2^{-m} \leq \frac{1}{6p} \delta(B_t) \leq \frac{\delta}{6} \nu_t(C),$$

where the first inequality follows from Definition 3(i), the second from (27), the third since $\delta(B_t) \geq 2^{-m}$, and the fourth from part (iii).

As this contradicts (26), we conclude that no such box $B$ exists, and hence $P(t)$ holds. The result then follows by induction. \qed

To prove our regret bound, we must now show Algorithm 1 selects good boxes $B_t$. In the following, we will say Algorithm 1 selects a box $B$ at time $t$ if $B_t = B$, and let $N_t(B)$ denote the number of times $s \leq t$ at which the box $B$ was selected. We then give the following lemma, which on a clean execution bounds how many different boxes we activate, and the number of times each box is selected.

**Lemma 7.** In the setting of Theorem 3, on a clean execution, for any $B \in B$ and $m \in \mathbb{N}$:

(i) at most $\kappa 2^{(m-1)\beta+1}$ boxes $B$ are activated at times $t \in \mathbb{N}$ with

$$2^{-m} \leq \delta(B_t) \leq 2^{1-m};$$

(ii) at most $\zeta_m$ activated boxes $B$ satisfy $\delta(B) \geq 2^{-m}$, where

$$\zeta_m = 1 + \sum_{l=0}^{m-1} \kappa 2^{l\beta+1};$$
(iii) if $B$ was activated, and $\delta(B) \geq 2^{-m}$, then $d(B) \leq \lambda m$; and

(iv) if $\Delta(B) \geq 2^{-m}$, then for any $T \in \mathbb{N}$

$$N_T(B)\Delta(B) \lesssim \gamma^{-1}2^m(m + \log(T)),$$

uniformly in $\gamma$.

Proof. We will show each result in turn, applying Lemma 6 as necessary.

(i) We will apply Lemma 6(iv). By that result, if a box $B$ was activated at a time $t \in \mathbb{N}$, with $2^{-m} \leq \delta(B_t) \leq 2^{-1-m}$, then $B$ is on the cover $C_m$. We therefore need only count the activated boxes $B$ on $C_m$.

The set of all activated boxes forms a tree, with root $X$, and $B$ a child of $C$ if $B$ was activated when $C$ was deactivated. The set of activated boxes on the cover $C_m$ similarly forms a forest, where each internal node has at least two children, and by Definition 2(ii) there are at most $\kappa 2^{(m-1)\beta}$ leaves. We conclude there are at most $\kappa 2^{(m-1)\beta+1}$ such boxes.

(ii) We will apply part (i). If $B \neq X$, it must have been activated at some time $t \in \mathbb{N}$. Since $B \subset B_t$, if $\delta(B) \geq 2^{-m}$, we have

$$\delta(B_t) \geq 2^{-m}.$$ 

By part (i), there are at most $\sum_{l=0}^{m-1} \kappa 2^{l\beta+1}$ such $B$; the result follows after including $B = X$.

(iii) As in parts (i) and (ii), $B$ must be on a cover $C_l$, for some $l \leq m$. The result then follows by Definition 2(iii).

(iv) We will apply Lemma 6(ii) and part (iii). If $N_T(B) = 0$, the result is trivial. Otherwise, $B$ must have been last selected at some time $t \leq T$. We then have

$$N_T(B) \leq n_t(B) \lesssim \nu^2 \Delta(B)^{-2} \log[\rho(B)(\tau + n_t(B))],$$

$$\lesssim \gamma^{-1} \Delta(B)^{-2} \log[\rho(B)T],$$

where the first inequality follows from the definitions, the second from Lemmas 6(ii) and 6(iv), and the third since $n_t(B) \leq T$. Now, since

$$\delta(B) \geq \Delta(B) \geq 2^{-m},$$

by part (iii) we also have $d(B) \leq \lambda m$, so $\log[\rho(B)] \lesssim m$, and the result follows. \qed
Finally, we will also prove a lemma controlling the size of the confidence radii \( r_t(B_t) \).

**Lemma 8.** In the setting of Theorem 3, on a clean execution, for any \( B \in \mathcal{B} \) and \( t \in \mathbb{N} \):

(i) if \( B \) was deactivated at time \( t \), then \( \delta(B) \geq \nu r_t(B) \);

(ii) if \( B \) was activated at time \( t \), then \( d(B) \lesssim \log(t) \); and

(iii) for any \( T \in \mathbb{N} \) and \( m \in \mathbb{Z} \), we have some \( t \leq T \) satisfying

\[ \nu r_t(B_t) \lesssim \max \left( 2^{-m}, \sqrt{\zeta_m \log(T) / \gamma T} \right), \]

uniformly in \( \gamma \).

**Proof.** We prove each result in turn, applying Lemma 7 where necessary.

(i) The result follows from the definitions. We have

\[ \delta(B) \geq W(B) \geq W_t(B) \geq \nu r_t(B), \]

where the first inequality follows from (10), the second from Definition 3(i), and the third since \( B \) was deactivated at time \( t \).

(ii) We proceed using part (i). Let \( C \) be the box deactivated immediately before \( B \) was activated; we have

\[ \delta(C) \geq \nu r_t(C) \gtrsim 1/\sqrt{\gamma t} \gtrsim 1/\sqrt{t}, \]

where the first inequality follows by part (i), the second since \( n_t(C) \leq t \), and the third since \( \gamma \leq 1 \). Then by Lemma 7(iii),

\[ d(C) \lesssim \log(t). \]

The result follows since \( d(B) \leq d(C) + 1 \).

(iii) We proceed using parts (i) and (ii). Given \( T \in \mathbb{N}, m \in \mathbb{Z} \), suppose that

\[ \nu r_t(B_t) \geq 2^{-m}, \quad t \leq T. \]

For any \( t \leq T \) such that \( B_t \neq X \), let \( C \) be the box deactivated immediately before \( B_t \) was activated, and \( s < t \) the time when this occurred. Since \( C \subseteq B_s \), we have

\[ \nu r_s(C) \geq \nu r_s(B_s) \geq 2^{-m}. \]

By part (i), we thus have \( \delta(C) \geq 2^{-m} \). By Lemma 7(ii), there are at most \( \zeta_m \) such boxes \( C \), and thus at most \( q \zeta_m + 1 \) distinct boxes \( B_t \), \( t \leq T \).
By the pigeonhole principle, we then must have some box $B$, last
selected at time $t \leq T$, for which
\[ n_t(B) = n_T(B) \geq T/(q\zeta_m + 1). \]

We deduce that
\[ \nu_t(B) \lesssim \sqrt{\zeta_m \log(\rho(B)(T + T))/\gamma T}, \]
and result follows by part (ii).

We are now ready to prove our regret bound for Algorithm 1.

\textbf{Proof of Theorem 3.} We will show that, on a clean execution, we can bound the regret of Algorithm 1 using Lemmas 6–8. Let $E_T$ be the event that the execution was clean; from Lemma 5, we know this event has probability at least $1 - \varepsilon$.

We next consider the cumulative regret. Set $\alpha = 1 + 1(\beta = 0)$, and given $T \in \mathbb{N}$, choose $m_T \in \mathbb{Z}$ so that
\[ 2^{m_T} \approx \left( \frac{\gamma T}{\log(T)^{\alpha}} \right)^{1/(\beta+2)}. \]

On $E_T$, for any set of reward distributions $P \in \mathcal{P}$, we have
\[
R_T = \sum_{B \text{ activated}} N_T(B)\Delta(B) \\
\leq \sum_{m=1}^{m_T} \sum_{\substack{B \text{ activated} \\ 2^{-m} \leq \Delta(B) \leq 2^{1-m}}} N_T(B)\Delta(B) + T2^{-m_T} \\
\lesssim \gamma^{-1} \sum_{m=1}^{m_T} \zeta_m 2^m (m + \log(T)) + T2^{-m_T} \\
\lesssim \gamma^{-1} \log(T)^{\alpha} 2^{(\beta+1)m_T} + T2^{-m_T} \\
\lesssim T \left( \frac{\gamma T}{\log(T)^{\alpha}} \right)^{-1/(\beta+2)},
\]
where the first inequality follows since we can select boxes with $\Delta(B) \leq 2^{-m_T}$ at most $T$ times, the second from Lemmas 7(ii) and 7(iv), and the third and fourth from the definitions of $\zeta_m$ and $m_T$; note the additional log power when $\beta = 0$ is necessary to control $\zeta_m$.
To bound the simple regret, we proceed in a similar fashion. On $E_T$, we have

$$S_T \leq \delta(B_{T_*}) \leq 5\nu r_T(B_{T_*}) \leq \max \left( 2^{-m_T}, \sqrt{\zeta m_T \log(T) / \gamma T} \right) \leq \max \left( 2^{-m_T}, \sqrt{2 \beta m_T \log(T) \alpha / \gamma T} \right) \leq \left( \frac{\gamma T}{\log(T)^\alpha} \right)^{-1/(\beta+2)},$$

where the first inequality follows from (10), the second from Lemmas 6(iii) and 6(iv), the third from Lemma 8(iii), and the fourth and fifth from the definitions of $\zeta_m$ and $m_T$.

### 4.4 Proofs of computational cost

Finally, we prove bounds on the complexity of our algorithm.

*Proof of Theorem 4.* We first prove our result in the setting of Theorem 3, and then deduce the result in the setting of Corollary 3.

(i) In the setting of Theorem 3, we work on the event $E_T$ that the execution was clean. We then divide the computational cost of Algorithm 1 into four parts:

(a) the cost of updating priorities in the priority queue;
(b) the cost of insertions to and deletions from the priority queue;
(c) the costs otherwise associated with maintaining Invariant 1; and
(d) all other costs.

For part (ia), we note that at time $t \leq T$, we only update the priority of the active box $B_t$. Since every deactivated box must contain at least one point $x_s$, there can be at most $O(T)$ active boxes. Each update therefore has cost $O(\log(T))$, and the total cost of updates is $O(T \log(T))$.

For part (ib), we note that the boxes activated by time $T$ form a tree, as in the proof of Lemma 7(i). As the leaves of the tree are the boxes active at time $T$, there are $O(T)$ leaves.

Since all internal nodes of the tree have at least two children, there are thus $O(T)$ boxes in total. Then as each box can have been inserted to
and deleted from the priority queue at most once, with cost $O(\log(T))$, the total cost of insertions and deletions is

$$O(T \log(T)).$$

For part (ic), let $\Gamma \in \mathbb{N}$ be an upper bound on the number of sub-boxes $C \subseteq B$ satisfying $\pi(C \mid B) \geq \gamma$, for any box $B$. Deactivating a box $B$ at time $t$ requires the computation of any newly-stored quantities: $O(\Gamma)$ such quantities taking time at most linear in $n_t(B)$; and the width estimates $W_t(C)$ of the newly-activated boxes $C$, which an efficient implementation can compute in time $O(\Gamma)$.

The total cost of activations is then

$$\sum_{B \text{ activated by time } T} O(\Gamma n_T(B)) = \sum_{t=1}^T \sum_{B \text{ hit by } x_t, \text{ and activated by time } T} O(\Gamma).$$

Now, by Lemma 8(ii), each design point $x_t$ can have hit at most $O(\log(T))$ boxes deactivated by time $T$. The total cost of activations is thus

$$O(\Gamma T \log(T)).$$

For part (id), we note that at time $t$, the remaining costs are those of updating stored quantities related to the selected box $B_t$: $O(\Gamma)$ stored quantities which can be updated in constant time; and $W_t(B_t)$, which can be updated in time $O(\Gamma)$. The total remaining cost thus

$$O(\Gamma T).$$

The overall computational cost is thus

$$O(\Gamma T \log(T)).$$

In the setting of Theorem 3, the result follows since $\Gamma$ is a fixed constant.

(ii) In the setting of Corollary 3, we note that our results need hold only in the limit $T \to \infty$. We may thus assume $T$ is large enough that $\log(T)^{-1} \leq \gamma$, and so apply the analysis in part (i).

To proceed, we need to bound the constant $\Gamma$ in terms of $T$. Choose $l \in \mathbb{N}$ as large as possible, such that $2^l \leq \log(T)$. Then for any box $B \in \mathcal{B}$, and any $C \subseteq B$ satisfying $\pi(C \mid B) \geq \log(T)^{-1}$, we must have that $C$ was constructed by splitting $B$ along the sequence of axes $i_1, \ldots, i_k$, for axes $i_j = 1, \ldots, p$, and $k \leq l$. 

46
There are $O(p^l)$ such sequences, and for each sequence, $O(\log(T))$ possible boxes $C$. We conclude that the total number of boxes $C$ is

$$\Gamma = O(p^l \log(T)) = O(\log(T)^{1+\log_2(p)}).$$

We thus obtain that the total computational cost is

$$O(T \log(T)^{2+\log_2(p)}).$$

Acknowledgements

We would like to thank Richard Nickl, Alexandra Carpentier, and the anonymous referees for their valuable comments and suggestions, and EPSRC for their support under grant EP/K000993/1.

References

Agrawal R. The continuum-armed bandit problem. *SIAM J. Control Optim.*, 33(6):1926–1951, 1995.

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

Auer P, Ortner R, and Szepesvári C. Improved rates for the stochastic continuum-armed bandit problem. In *Learning Theory*, volume 4539 of *Lecture Notes in Comput. Sci.*, pages 454–468. Springer, Berlin, 2007.

Bubeck S. *Jeux de bandits et fondations du clustering*. PhD thesis, Université Lille 1, 2010.

Bubeck S, Stoltz G, and Yu J. Lipschitz bandits without the Lipschitz constant. In *Algorithmic Learning Theory*, pages 144–158. Springer, 2011a.

Bubeck S and Cesa Bianchi N. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *arXiv preprint arXiv:1204.5721*, 2012.

Bubeck S, Munos R, and Stoltz G. Pure exploration in multi-armed bandits problems. In *Proc. 20th International Conference on Algorithmic Learning Theory (ALT ’09)*, pages 23–37, Porto, Portugal, 2009.

Bubeck S, Munos R, Stoltz G, and Szepesvári C. $X$-armed bandits. *J. Mach. Learn. Res.*, 12:1655–1695, 2011b.

Cope E W. Regret and convergence bounds for a class of continuum-armed bandit problems. *IEEE Trans. Automat. Control*, 54(6):1243–1253, 2009.

Coquelin P and Munos R. Bandit algorithms for tree search. *arXiv preprint cs/0703062*, 2007.

Frazier P, Powell W, and Dayanik S. The knowledge-gradient policy for correlated normal beliefs. *INFORMS J. Comput.*, 21(4):599–613, 2009.

Gelly S, Kocsis L, Schoenauer M, Sebag M, Silver D, Szepesvári C, and Teytaud O. The grand challenge of computer Go: Monte Carlo tree search and extensions. *Communications of the ACM*, 55(3):106–113, 2012.
Huang D, Allen T T, Notz W I, and Miller R A. Sequential kriging optimization using multiple-fidelity evaluations. *Struct. Multidiscip. Optim.*, 32(5):369–382, 2006.

Kiefer J and Wolfowitz J. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 23(3):462–466, 1952.

Kleinberg R, Slivkins A, and Upfal E. Multi-armed bandits in metric spaces. In *Symposium on Theory of Computing ’08*, pages 681–690. ACM, New York, 2008.

Kleinberg R D. Nearly tight bounds for the continuum-armed bandit problem. In *Advances in Neural Information Processing Systems 17*, pages 697–704. MIT Press, Cambridge, MA, 2005.

Kocsis L and Szepesvári C. Bandit based Monte-Carlo planning. In *European Conference on Machine Learning 17*, volume 4212 of *Lecture Notes in Comput. Sci.*, pages 282–293. Springer, Berlin, 2006.

Munos R. Optimistic optimization of a deterministic function without the knowledge of its smoothness. In *Advances in Neural Information Processing Systems 24*, pages 783–791. MIT Press, Cambridge, MA, 2011.

Pandey S, Agarwal D, Chakrabarti D, and Josifovski V. Bandits for taxonomies: a model-based approach. In *SIAM Intl. Conf. on Data Mining (SDM)*, 2007.

Parsopoulos K E and Vrahatis M N. Recent approaches to global optimization problems through particle swarm optimization. *Natural computing*, 1(2-3):235–306, 2002.

Robbins H. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

Slivkins A. Multi-armed bandits on implicit metric spaces. In *Advances in Neural Information Processing Systems 24*, pages 1602–1610. MIT Press, Cambridge, MA, 2011.

Srinivas N, Krause A, Kakade S M, and Seeger M. Gaussian process optimization in the bandit setting: no regret and experimental design. In *International Conference on Machine Learning*, Haifa, Israel, 2010.

Valko M, Carpentier A, and Munos R. Stochastic simultaneous optimistic optimization. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, page 19–27, 2013.

Yu J Y and Mannor S. Unimodal bandits. In *Proceedings of the 28th International Conference on Machine Learning*, 2011.