Exponential Regret Bounds for Gaussian Process Bandits with Deterministic Observations

Nando de Freitas NANDO@CS.UBC.CA
Department of Computer Science, University of British Columbia, Vancouver, BC V6T 1Z4, Canada

Alex J. Smola ALEX@SMOLA.ORG
Yahoo! Research, Santa Clara, CA 95051, USA

Masrour Zoghi MZOGHI@CS.UBC.CA
Department of Computer Science, University of British Columbia, Vancouver, BC V6T 1Z4, Canada

Abstract

This paper analyzes the problem of Gaussian process (GP) bandits with deterministic observations. The analysis uses a branch and bound algorithm that is related to the UCB algorithm of (Srinivas et al., 2010). For GPs with Gaussian observation noise, with variance strictly greater than zero, (Srinivas et al., 2010) proved that the regret vanishes at the approximate rate of $\mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$, where t is the number of observations. To complement their result, we attack the deterministic case and attain a much faster exponential convergence rate. Under some regularity assumptions, we show that the regret decreases asymptotically according to $\mathcal{O}\left(e^{-\frac{\tau t}{(\ln t)^{d/4}}}\right)$ with high probability. Here, d is the dimension of the search space and τ is a constant that depends on the behaviour of the objective function near its global maximum.

1. Introduction

Let $f : \mathcal{D} \to \mathbb{R}$ be a function on a compact subset $\mathcal{D} \subseteq \mathbb{R}^d$. We would like to address the global optimization problem

$$x_M = \operatorname*{argmax}_{x \in \mathcal{D}} f(x).$$

Let us assume for the sake of simplicity that the objective function f has a unique global maximum (although it may have many local maxima).

The space $\mathcal{D}$ might be the set of free parameters that one could feed into a time-consuming algorithm or the locations where a sensor could be deployed, and the function f might be a measure of the performance of the algorithm (e.g. how long it takes to run). We refer the reader to (Močkus, 1982; Schonlau et al., 1998; Gramacy et al., 2004; Brochu et al., 2007; Lizotte, 2008; Martinez–Cantin et al., 2009; Garnett et al., 2010) for practical examples. In this paper, our assumption is that once the function has been probed at point $x \in \mathcal{D}$, then the value $f(x)$ can be observed with very high precision. This is the case when the deployed sensors are very accurate or if the algorithm is deterministic. An example of this is the configuration of CPLEX parameters in mixed-integer programming (Hutter et al., 2010). More ambitiously, we might be interested in the *simultaneous* automatic configuration of an entire system (algorithms, architectures and hardware) whose performance is deterministic in terms of several free parameters and design choices.

Global optimization is a difficult problem without any assumptions on the objective function f. The main complicating factor is the uncertainty over the extent of the variations of f, e.g. one could consider the characteristic function, which is equal to 1 at x_M and 0 elsewhere, and none of the methods we mention here can optimize this function without exhaustively searching through every point in $\mathcal{D}$.

The way a large number of global optimization methods address this problem is by imposing some prior assumption on how fast the objective function f can vary. The most explicit manifestation of this remedy is the imposition of a Lipschitz assumption

Appearing in *Proceedings of the 29th International Conference on Machine Learning*, Edinburgh, Scotland, UK, 2012. Copyright 2012 by the author(s)/owner(s).

on f, which requires the change in the value of $f(x)$, as the point x moves around, to be smaller than a constant multiple of the distance traveled by x (Hansen et al., 1992). As pointed out in (Bubeck et al., 2011, Figure 3), it is only important to have this kind of tight control over the function near its optimum: elsewhere in the space, we can have what they have dubbed a "weak Lipschitz" condition.

One way to relax these hard Lipschitz constraints is by putting a Gaussian Process (GP) prior on the function. Instead of restricting the function from oscillating too fast, a GP prior requires those fast oscillations to have low probability, cf. (Ghosal & Roy, 2006, Theorem 5).

The main point of these bounds (be they hard or soft) is to assist with the *exploration-exploitation trade-off* that global optimization algorithms have to grapple with. In the absence of any assumptions of convexity on the objective function, a global algorithm is forced to explore enough until it reaches a point in the process when with some degree of certainty it can localize its search space and perform local optimization (exploitation). Derivative bounds such as the ones discussed here together with the boundedness of the search space, guaranteed by the compactness assumption on $\mathcal{D}$, provide us with such certainty by producing a useful upper bound that allows us to shrink the search space. This is illustrated in Figure 1. Suppose we know that our function is Lipschitz with constant L, then given sample points as shown in the figure, we can use the Lipschitz property to discard pieces of the search space. This is done by finding points in the search space where the function could not possibly be higher than the maximum value already encountered. Such points are found by placing cones at the sampled points with slope equal to L and checking where those cones lie below the maximum observed value.

This crude approach is wasteful because very often the slope of the function is much smaller than L. As shown in Figure 2), GPs do a better job of providing lower and upper bounds that can be used to limit the search space, by essentially choosing Lipschitz constants that vary over the search space and the algorithm run time.

We also assume that the objective function f is costly to evaluate. We would like to avoid probing f as much as possible and to get close to the optimum as quickly as possible. A solution to this problem is to approximate f with a *surrogate function* that provides a good upper bound for f and which is easier to calculate and optimize (Brochu et al., 2009). Surrogate functions also aid with global optimization by restricting the domain of interest. The surrogate that we will make extensive use of here is called the Upper Confidence Bound (UCB). It is defined to be $\mu + B\sigma$, where μ and σ are the posterior predictive mean and standard deviation of the GP and B is a constant to be chosen by the algorithm. This surrogate function has been studied extensively in the literature and this paper relies heavily on the ideas put forth in the paper by Srinivas et al (Srinivas et al., 2010), in which the algorithm consists of repeated optimization of the UCB surrogate function after each sample. It must be noted however that our algorithm is distinctly different from their UCB algorithm.

One key difference between our setting and that of (Srinivas et al., 2010) is that, whereas we assume that the value of the function can be observed exactly, for the analysis presented in (Srinivas et al., 2010) it is necessary for the noise to be non-trivial (and Gaussian) because the main quantity that is used in the estimates, namely information gain, cf. (Srinivas et al., 2010, Equation 3), becomes undefined when the variance of the observation noise (σ^2 in their notation) is set to 0, cf. the expression for $\mathrm{I}(\mathbf{y}_A; \mathbf{f}_A)$ that was given in the paragraph following Equation (3). So, our analysis is complementary to theirs. Of course, one could still use their algorithm in the noiseless setting, but their analytical results are inapplicable to that case. Moreover, we show that the regret, $r(x_t) = \max_{\mathcal{D}} f - f(x_t)$, decreases according to $\mathcal{O}\left(e^{-\frac{\tau t}{(\ln t)^{d/4}}}\right)$, implying that the cumulative regret is bounded from above.

The paper whose results are most similar to ours is (Munos, 2011), but there are some key differences in the methodology, analysis and obtained rates. For instance, we are interested in cumulative regret, whereas the results of (Munos, 2011) are proven for finite stop-time regret. In our case, the ideal application is the optimization of a function that is C^2-smooth and has an unknown non-singular Hessian at the maximum. We obtain a regret rate $\mathcal{O}\left(e^{-\frac{\tau t}{(\ln t)^{d/4}}}\right)$, whereas the DOO algorithm in (Munos, 2011) has regret rate $\mathcal{O}(e^{-t})$ if the Hessian is known and the SOO algorithm has regret rate $\mathcal{O}(e^{-\sqrt{t}})$ if the Hessian is unknown. In addition, the algorithms in (Munos, 2011) can handle functions that behave like $-c\|x - x_M\|^{\alpha}$ near the maximum (cf. Example 2 therein). Moreover, the hierarchical decomposition of the search space utilized by DOO and SOO makes them much more efficient in practice than the algorithm presented in

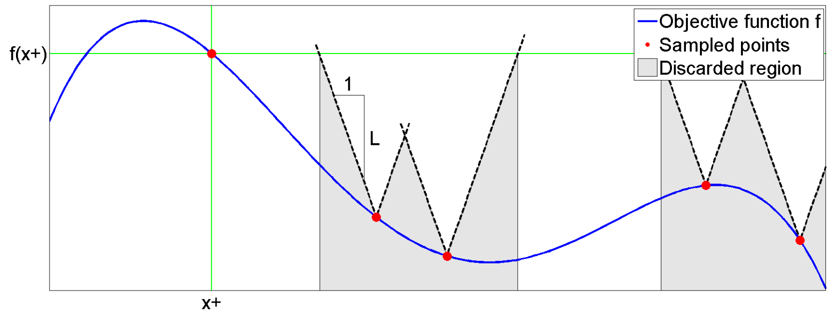


Figure 1. An example of the Lipschitz hypothesis being used to discard pieces of the search space when finding the maximum of a function f. Although f is only known at the red sample points, if the derivative upper bounds (dashed lines) are below the best attained value thus far, $f(x^+)$, the corresponding areas of the search space (shaded regions) may be discarded.

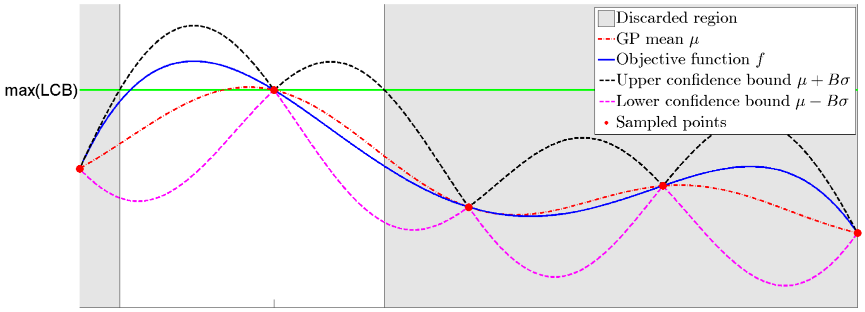


Figure 2. An example of our branch and bound maximization algorithm with UCB surrogate $\mu + B\sigma$, where μ and σ are the mean and standard deviation of the GP respectively. The region consisting of the points x for which the upper confidence bound $\mu(x) + B\sigma(x)$ is lower that the maximum value of the lower confidence bound $\mu(x) - B\sigma(x)$ does not need to be sampled anymore. Note that the UCB surrogate function bounds f from above.

this paper: this is a shortcoming of our algorithm that we would like to remedy in the future.

This problem was also studied by (Vazquez & Bect, 2010) and (Bull, 2011), but using the Expected Improvement surrogate instead of UCB. Our methodology and results are different, but complementary to theirs.

2. Gaussian process bandits

2.1. Gaussian processes

As in (Srinivas et al., 2010), the objective function is distributed according to a Gaussian process prior:

$$f(x) \sim \mathrm{GP}(m(\cdot), \kappa(\cdot, \cdot)). \tag{1}$$

For convenience, and without loss of generality, we assume that the prior mean vanishes, i.e., $m(\cdot) = 0$. There are many possible choices for the covariance kernel. One obvious choice is the anisotropic kernel κ with a vector of known hyperparameters (Rasmussen & Williams, 2006):

$$\kappa(x_i, x_j) \quad = \quad \widetilde{\kappa}\left(-(x_i - x_j)^\top \mathbf{D}(x_i - x_j)\right), \tag{2}$$

where $\widetilde{\kappa}$ is an isotropic kernel and $\mathbf{D}$ is a diagonal matrix with positive hyperparameters along the diagonal and zeros elsewhere. Our results apply to squared exponential kernels and Matérn kernels with parameter $\nu \geq 2$. In this paper, we assume that the hyperparameters are fixed and known in advance.

We can sample the GP at t points by choosing points $\mathbf{x}_{1:t} := \{x_1, \ldots, x_t\}$ and sampling the values of the function at these points to produce the vector $\mathbf{f}_{1:t} = [f(x_1) \cdots f(x_t)]^\top$. The function values are distributed according to a multivariate Gaussian distribution $\mathcal{N}(0, \mathbf{K})$, with covariance entries $\kappa(x_i, x_j)$. Assume that we already have several observations from previous steps, and that we want to decide what action x_{t+1} should be considered next. Let us denote the value of the function at this arbitrary new point as f_{t+1}. Then, by the properties of GPs,

$\mathbf{f}_{1:t}$ and f_{t+1} are jointly Gaussian:

$$\begin{bmatrix}\mathbf{f}_{1:t}\\ f_{t+1}\end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix}\mathbf{K} & \mathbf{k}^\top\\ \mathbf{k} & \kappa(x_{t+1}, x_{t+1})\end{bmatrix}\right),$$

where $\mathbf{k} = [\kappa(x_{t+1}, x_1) \cdots \kappa(x_{t+1}, x_t)]^\top$. Using the Schur complement, one arrives at an expression for the posterior predictive distribution:

$$P(f_{t+1}|\mathbf{x}_{1:t+1}, \mathbf{f}_{1:t}) = \mathcal{N}(\mu_t(x_{t+1}), \sigma_t^2(x_{t+1})),$$

where

$$\begin{aligned}\mu_t(x_{t+1}) &= \mathbf{k}^\top \mathbf{K}^{-1}\mathbf{f}_{1:t},\\ \sigma_t^2(x_{t+1}) &= \kappa(x_{t+1}, x_{t+1}) - \mathbf{k}^\top \mathbf{K}^{-1}\mathbf{k}\end{aligned} \tag{3}$$

and $\mathbf{f}_{1:t} = [f(x_1) \cdots f(x_t)]^\top$.

2.2. Surrogates for optimization

When it is assumed that the objective function f is sampled from a GP, one can use a combination of the posterior predictive mean and variance given by Equations (3) to construct surrogate functions, which tell us where to sample next. Here we use the UCB combination, which is given by

$$\mu_t(x) + B_t\sigma_t(x),$$

where $\{B_t\}_{t=1}^\infty$ is a sequence of numbers specified by the algorithm. This surrogate trades-off exploration and exploitation since it is optimized by choosing points where the mean is high (exploitation) and where the variance is large (exploration). Since the surrogate has an analytical expression that is easy to evaluate, it is much easier to optimize than the original objective function. Other popular surrogate functions constructed using the sufficient statistics of the GP include the Probability of Improvement, Expected Improvement and Thompson sampling. We refer the reader to (Brochu et al., 2009; May et al., 2010; Hoffman et al., 2011) for details on these.

2.3. Our algorithm

The main idea of our algorithm (Algorithm 1) is to tighten the bound on f given by the UCB surrogate function by sampling the search space more and more densely and shrinking this space as more and more of the UCB surrogate function is "submerged" under the maximum of the Lower Confidence Bound (LCB). Figure 2 illustrates this intuition.

More specifically, the algorithm consists of two iterative stages. During the first stage, the function is sampled at enough points in $\mathcal{L}$ (the red crosses in Figure 3) until every point in the search space is contained inside a simplex of diameter δ, where by the diameter of a set we mean the maximum length between any pair of points in the set. In the second stage, the search space is shrunk to discard regions where the maximum is very unlikely to reside. Such regions are obtained by finding points where the UCB is lower than the LCB (the complement of the colored region in the same panel as before). The remaining set of relevant points is denoted by $\widetilde{\mathcal{R}}$. To simplify the task of shrinking the search space, we simply find an enclosing ball, which is denoted by $\mathcal{R}$ in Algorithm 1. Back to the first stage, we consider a lattice that is twice as dense as in the first stage of the previous iteration, but we only sample at points that lie within our new smaller search space.

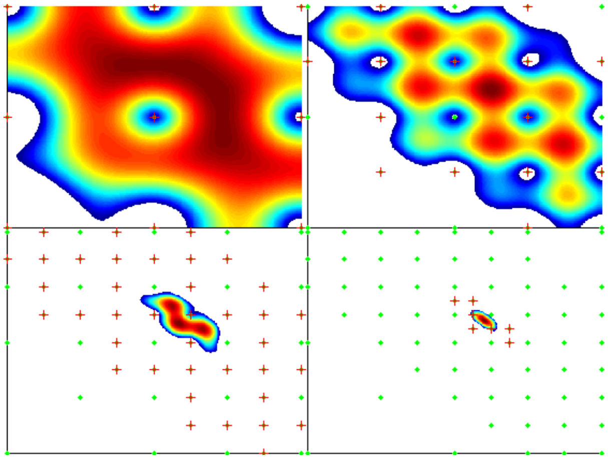


Figure 3. Branch and Bound algorithm for a 2D function. The colored region is the search space and the colormap, with red high and blue low, illustrates the value of the UCB. Four steps of the algorithm are shown; progressing from left to right and top to bottom. The green dots designate the points where the function was sampled in the previous steps, while the red crosses denote the freshly sampled points.

In the second stage, the auxiliary step of approximating the relevant set $\widetilde{\mathcal{R}}$ with the ball $\mathcal{R}$ introduces inefficiencies in the algorithm, since we only need to sample inside $\widetilde{\mathcal{R}}$. This can be easily remedied in practice to obtain an efficient algorithm. Our analysis will show that even without these improvements it is already possible to obtain very strong exponential convergence rates. Of course, practical improvement will result in better constants and ought to be considered seriously.

Note that Algorithm 1 terminates once the relevant region becomes too small to intersect the lattice $\mathcal{L}$. Our analysis requires for the algorithm to sample

Algorithm 1 Branch and Bound

```
Input: A compact subset $\mathcal{D} \subseteq \mathbb{R}^d$, a function $f : \mathcal{D} \to \mathbb{R}$ and a discrete lattice $\mathcal{L} \subseteq \mathcal{D}$ that is divisible by powers of 2. Set $\mathcal{R} \leftarrow \mathcal{D}$ and $\delta \leftarrow 1$.
repeat
  Sample Twice as Densely:
    • $\delta \leftarrow \delta/2$.
    • Sample $f$ at enough points in $\mathcal{L}$ so that every point in $\mathcal{R}$ is contained in a simplex of diameter $\delta$.
  Shrink the Relevant Region:
    • Set
        $\widetilde{\mathcal{R}} := \left\{ x \in \mathcal{R} \middle| \mu_T(x) + \sqrt{\beta_T}\sigma_T(x) > \sup_{\mathcal{R}} \mu_T(x) - \sqrt{\beta_T}\sigma_T(x) \right\}.$
      $T$ is the number points sampled so far and $\beta_T = 2\ln\left(\frac{|\mathcal{L}|T^2}{\alpha}\right) = 4\ln T + 2\ln\frac{|\mathcal{L}|}{\alpha}$ with $\alpha \in (0,1)$.
    • Solve the following constrained optimization problem: $(x_1^*, x_2^*) = \operatorname{argsup}_{(x_1,x_2)\in\widetilde{\mathcal{R}}\times\widetilde{\mathcal{R}}} \|x_1 - x_2\|$.
    • $\mathcal{R} \leftarrow B\left(\frac{x_1^* + x_2^*}{2}, \|x_1^* - x_2^*\|\right)$, where $B(p,r)$ is the ball of radius $r$ centred around $p$.
until $\mathcal{R} \cap \mathcal{L} = \varnothing$
```

points from a fixed finite set of points, although we can pick $\mathcal{L}$ to be the set of all points in $\mathcal{D}$ with floating point coordinates.

3. Analysis

We begin our analysis by showing that, given sufficient explored locations, the posterior predictive variance is small. Specifically, the following approximation result is proved in the supplementary material:

Proposition 1 (Variance Bound) *Let $\kappa : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a kernel that is four times differentiable along the diagonal $\{(x,x) \,|\, x \in \mathbb{R}^d\}$, with Q defined as in part 2 of Lemma 5, and $f \sim \mathrm{GP}\left(0, \kappa(\cdot,\cdot)\right)$ a sample from the corresponding GP. If f is sampled at points $x_{1:T} = \{x_1, \ldots, x_T\}$ that form a δ-cover of a subset $\mathcal{D} \subseteq \mathbb{R}^d$, then the resulting posterior predictive standard deviation σ_T satisfies*

$$\sup_{\mathcal{D}} \sigma_T \leq \frac{Q\delta^2}{4}.$$

3.1. Finiteness of regret

Having shown that the variance vanishes according to the square of the resolution of the lattice of sampled points, we now move on to show that this estimate implies an exponential asymptotic vanishing of the regret encountered by our Branch and Bound algorithm. This is laid out in our main theorem stated below and proven in the supplementary material.

Recall that $\mathcal{D} \subseteq \mathbb{R}^d$ is assumed to be a non-empty compact subset and f a sample from the GP $\mathrm{GP}\left(0, \kappa(\cdot,\cdot)\right)$ on $\mathcal{D}$. Moreover, in what follows we will denote the global maximum by $x_M := \operatorname{argmax}_{x\in\mathcal{D}} f(x)$ and the regret by $r(x_t) = f(x_M) - f(x_t)$. Also, by convention, for any set $\mathcal{S}$, we will denote its interior by $\mathcal{S}^\circ$, its boundary by $\partial\mathcal{S}$ and if S is a subset of $\mathbb{R}^d$, then $\mathrm{conv}(S)$ will denote its convex hull. The following holds true:

Theorem 2 *Suppose we are given:*

1. *$\alpha > 0$, a compact subset $\mathcal{D} \subseteq \mathbb{R}^d$, and κ a kernel on $\mathbb{R}^d$ that is four times differentiable along the diagonal;*
2. *$f \sim \mathrm{GP}(0,\kappa)$ a continuous sample on $\mathcal{D}$ that has a unique global maximum x_M, which satisfies one of the following two conditions:*
   - (†) *$x_M \in \mathcal{D}^\circ$ and $f(x_M) - c_1\|x - x_M\|^2 < f(x) \leq f(x_M) - c_2\|x - x_M\|^2$ for all x satisfying $x \in B(x_M, \rho_0)$ for some $\rho_0 > 0$;*
   - (‡) *$x_M \in \partial\mathcal{D}$ and both f and $\partial\mathcal{D}$ are smooth at x_M, with $\nabla f(x_M) \neq 0$;*
3. *any lattice $\mathcal{L} \subseteq \mathcal{D}$ satisfying the following two conditions*
   - $$2\mathcal{L} \cap \mathrm{conv}(\mathcal{L}) \subseteq \mathcal{L} \tag{4}$$
   - $$2^{\left\lceil -\log_2 \frac{\rho_0}{\mathrm{diam}(\mathcal{D})} \right\rceil + 1}\mathcal{L} \cap \mathcal{L} \neq \varnothing \tag{5}$$

     if f satisfies (†)

Then, there exist positive numbers A and τ and an integer T such that the points specified by the Branch and Bound algorithm, $\{x_t\}$, will satisfy the following asymptotic bound: For all $t > T$, with probability $1 - \alpha$ we have

$$r(x_t) < Ae^{-\frac{\tau t}{(\ln t)^{d/4}}}.$$

Given the exponential rate of convergence we obtain in Theorem 2, we have the following finiteness conclusion for the cumulative regret accrued by our Branch and Bound algorithm:

Corollary 3 *Given κ, $f \sim \mathrm{GP}(0, \kappa)$ and $\mathcal{L} \subseteq \mathcal{D}$ as in Theorem 2, the cumulative regret is bounded from above.*

Remark 4 *It is worth pointing out the trivial observation that using a simple UCB algorithm with monotonically increasing and unbounded factor $\sqrt{\beta_t}$, without any shrinking of the search space as we do here, necessarily leads to unbounded cumulative regret since eventually $\sqrt{\beta_t}$ becomes large enough so that at points x' far away from the maximum, $\sqrt{\beta_t}\sigma_t(x')$ becomes larger than $f(x_M) - f(x)$. In fact, eventually the UCB algorithm will sample every point in the lattice $\mathcal{L}$.*

3.2. Remarks on the main theorem

This section includes a discussion of the assumptions placed on the objective function in Theorem 2 as well as an outline of the proof, the full details of which are included in the appendix.

3.2.1. On the statement of Theorem 2

A few remarks on the assumptions and the conclusion of the main theorem are in order:

A. Relationship between the local and global assumptions on f: The theorem has two seemingly unrelated restrictions on the function f: the global GP prior and the local behaviour near the global maximum. However, in many circumstances of interest, the local condition follows almost surely from the global condition. Two such circumstances are if κ is a Matérn kernel with $\nu > 2$ (including the squared exponential kernel) or if κ is six times differentiable. In either case, the sample f is twice differentiable almost surely, in the former case by (Adler & Taylor, 2007, Theorem 1.4.2) and (Stein, 1999, §2.6)) and in the latter situation by (Ghosal & Roy, 2006, Theorem 5). If the global maximum x_M lies in the interior of $\mathcal{D}$, the Hessian of f at x_M will almost surely be non-singular since the vanishing of at least one of the eigenvalues of the Hessian is a codimension 1 condition in the space of all functions that are smooth at a given point, hence justifying condition (†).

On the other hand, if x_M lies on the boundary of $\mathcal{D}$, then condition (‡) will be satisfied almost surely, since the additional event of the vanishing of $\nabla f(x_M)$ is a codimension d phenomenon in the space of functions with global maximum at x_M.

B. Uniqueness of the global maximum: A randomly drawn continuous sample from a GP on a compact domain will almost surely have a unique global maximum: this is because the space of continuous functions on a compact domain that attain their global maximum at more than one point have codimension one in the space of all continuous functions on that domain.

C. Assumptions on $\mathcal{L}$: The two conditions (4) and (5) simply require that the lattice be "divisible by 2" and that it be fine enough so that the algorithm can sample inside the ball $B(x_M, \rho_0)$ when the maximum of the function is located in the interior of the search space $\mathcal{D}$. One can simply choose $\mathcal{L}$ to be the set of points in $\mathcal{D}$ that have floating point coordinates: it's just the points at which the algorithm is allowed to sample the function.

D. On τ's dependence: Finally, it is important to point out that the decay rate τ does not depend on the choice of the lattice $\mathcal{L}$, even though as stated, the statement of the theorem chooses τ only after $\mathcal{L}$ is specified. The theorem was written this way simply for the sake of readability.

3.2.2. Outline of the proof of Theorem 2

The starting point for the proof is the observation that one can use the posterior predictive mean and standard deviation of the GP to obtain a high probability envelope around the objective function (cf. Lemma 8 in the appendix). Given the fact that the thickness of this envelope is determined by the height of the posterior predictive standard deviation, σ, we can use the bound given by Proposition 1 to show that asymptotically one can rapidly dispense with large portions of the search space, as illustrated in Figure 2.

One disconcerting component of Algorithm 1 is the step that requires sampling twice as densely in each iteration, since the number of samples can start to grow exponentially, hence killing any hope of obtaining exponentially decreasing regret. However, this is where the assumption on the local behaviour near the global maximum becomes relevant. Since Proposition 1 tells us that every time the function is sampled twice as densely, σ decreases by a factor of 4, and given our assumption that the function has quadratic behaviour near the global maximum, we can conclude that the radius of the search space is halved after each iteration and so the number of sampled points added in each iteration roughly remains constant. Of course, this assumes that the multiplicative factor $\sqrt{\beta_t}$ remains constant in this process. However, the algorithm requires $\sqrt{\beta_t}$ to grow logarithmically, and so to fill this gap we need to bound the growth of $\sqrt{\beta_t}$, which is tied to the number of samples needed in each iteration of the algorithm, which in turn is linked to the resolution of the lattice of sampled points δ and the size of the relevant set $\mathcal{R}$, which in turn depends on the size of $\sqrt{\beta_t}\sigma_t$. This circular dependence gives rise to a difference equation, whose solutions we bound by solving the corresponding differential equation.

3.2.3. Further remarks on the GP prior

Let us step back for a moment and pose the question of whether it would be possible to carry out a similar line of reasoning under other circumstances. To answer this, one needs to identify the key ingredients of the proof, which are the following:

A. A mechanism for calculating a high probability envelope around the objective function (cf. Lemma 8);

B. An estimate showing that the thickness of the envelope diminishes rapidly as the function is sampled more and more densely (cf. Proposition 1), so that the search space can be shrunk under reasonable assumptions on the behaviour of the function near the peak.

The reason for our imposing a GP prior on f is that it gives us property A, while our smoothness assumption on the kernel guarantees property B. However, GPs are but one way one could obtain these properties and they do this essentially by coming up with local estimates of the Lipschitz constant based on the observed values of the objective function nearby. Perhaps one could explicitly incorporate similar local estimates on the Lipschitz constant into tree based approaches like HOO and SOO, cf. (Bubeck et al., 2011) and (Munos, 2011), in which case one would be able to dispense with the GP assumption and get similar performance. But, that is beyond the scope of this paper and will be left for future work.

4. Discussion

In this paper we proposed a modification of the UCB algorithm of (Srinivas et al., 2010) which addresses the noise free case. The key difference is that while the original algorithm achieves an $O(t^{-\frac{1}{2}})$ rate of convergence to the regret minimizer, we obtain an exponential rate in the number of function evaluations. In other words, the noise free problem is significantly easier, statistically speaking, than the noisy case. The key difference is that we need not invest any samples in noise reduction to determine whether our observations deviate far from their expectation.

This allows us to discard pieces of the search space where the maximum is very unlikely to be, when compared to (Srinivas et al., 2010). We show that this additional step leads to a considerable improvement of the regret accrued by the algorithm. In particular, the cumulative regret obtained by our Branch and Bound algorithm is bounded from above, whereas the cumulative regret bound obtained in the noisy bandit algorithm is unbounded. The possibility of dispensing with chunks of the search space can also be seen in the works involving hierarchical partitioning, e.g. (Munos, 2011), where regions of the space are deemed as less worthy of probing as time goes on.

Our results mirror the observation in active learning that noise free and large margin learning of half spaces can be achieved much more rapidly than identifying a linear separator in the noisy case (Bshouty & Wattad, 2006; Dasgupta et al., 2009). This is also reflected in classical uniform convergence results for supervised learning (Audibert & Tsybakov, 2007; Vapnik, 1998) where the achievable rate depends on the decay of probability mass near the margin.

This suggests that the ability to extend our results to the noisy case is somewhat limited. An indication of what might be possible can be found in (Balcan et al., 2009), where regions of the version space are eliminated once they can be excluded with sufficiently high probability. One could model a corresponding Branch and Bound algorithm, which dispenses with points that lie outside the current (or perhaps the previous) relevant set when calculating the covariance matrix $\mathbf{K}$ in the posterior equations (3). Analysis of how much of an effect such a computational cost-cutting measure would have on the regret encountered by the algorithm is a subject of future research.

Acknowledgements

We are very grateful to the anonymous reviewers for outstanding feedback. This research was supported by NSERC and the Institute for Computing, Information and Cognitive Systems (ICICS) at UBC.

References

Adler, R. J. and Taylor, J. E. *Random Fields and Geometry*. Springer, 2007.

Audibert, J.-Y. and Tsybakov, A. B. Fast learning rates for plug-in classifiers. *Annals of Statistics*, 35(2):608–633, 2007.

Balcan, M.-F., Beygelzimer, A., and Langford, J. Agnostic active learning. *J. Comput. Syst. Sci*, 75 (1):78–89, 2009.

Brochu, E., de Freitas, N., and Ghosh, A. Active preference learning with discrete choice data. In *NIPS*, pp. 409–416, 2007.

Brochu, E., Cora, V. M., and de Freitas, N. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. Technical Report TR-2009-023, arXiv:1012.2599v1, UBC CS department, 2009.

Bshouty, N. H. and Wattad, E. On exact learning halfspaces with random consistent hypothesis oracle. In *International Conference on Algorithmic Learning Theory*, pp. 48–62, 2006.

Bubeck, S., Munos, R., Stoltz, G., and Szepesvari, C. X-armed bandits. *Journal of Machine Learning Research*, 12:1655–1695, 2011.

Bull, A. D. Convergence rates of efficient global optimization algorithms. *Journal of Machine Learning Research*, 12:2879–2904, 2011.

Dasgupta, S., Kalai, A. T., and Monteleoni, C. Analysis of perceptron-based active learning. *Journal of Machine Learning Research*, 10:281–299, 2009.

Garnett, R., Osborne, M. A., and Roberts, S. J. Bayesian optimization for sensor set selection. In *ACM/IEEE International Conference on Information Processing in Sensor Networks*, pp. 209–219. ACM, 2010.

Ghosal, S. and Roy, A. Posterior consistency of Gaussian process prior for nonparametric binary regression. *Ann. Stat.*, 34:2413–2429, 2006.

Gramacy, R. B., Lee, H. K. H., and MacReady, W. Parameter space exploration with Gaussian process trees. In *ICML*, pp. 353–360, 2004.

Hansen, P., Jaumard, B., and Lu, S. Global optimization of univariate Lipschitz functions: I. survey and properties. *Mathematical Programming*, 55:251–272, 1992.

Hoffman, M., Brochu, E., and de Freitas, N. Portfolio allocation for Bayesian optimization. In *UAI*, pp. 327–336, 2011.

Hutter, F., Hoos, H. H., and Leyton-Brown, K. Automated configuration of mixed integer programming solvers. In *Proceedings of CPAIOR-10*, pp. 186–202, 2010.

Lizotte, D. *Practical Bayesian Optimization*. PhD thesis, University of Alberta, Canada, 2008.

Martinez–Cantin, R., de Freitas, N., Brochu, E., Castellanos, J., and Doucet, A. A Bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot. *Autonomous Robots*, 27(2):93–103, 2009.

May, B., Korda, N., Lee, A., and Leslie, D. Optimistic Bayesian sampling in contextual-bandit problems. 2010.

Močkus, J. The Bayesian approach to global optimization. In *System Modeling and Optimization*, volume 38, pp. 473–481. Springer, 1982.

Munos, R. Optimistic optimization of a deterministic function without the knowledge of its smoothness. In *NIPS*, 2011.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

Schonlau, M., Welch, W. J., and Jones, D. R. Global versus local search in constrained optimization of computer models. *Lecture Notes-Monograph Series*, 34:11–25, 1998.

Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. Gaussian process optimization in the bandit setting: No regret and experimental design. In *ICML*, 2010.

Stein, M. L. *Interpolation of Spatial Data: Some Theory for Kriging*. Springer, 1999.

Steinwart, I. and Christmann, A. *Support Vector Machines*. Springer, 2008.

Vapnik, V. *Statistical Learning Theory*. John Wiley and Sons, New York, 1998.

Vazquez, E. and Bect, J. Convergence properties of the expected improvement algorithm with fixed mean and covariance functions. *J. of Statistical Planning and Inference*, 140:3088–3095, 2010.

5. Proofs

5.1. Approximation Results

We begin by relating posterior predictive variance, projection operators, and interpolation in Hilbert Spaces. Lemmas 5, 6 and 7 are standard. We include their proofs for the purpose of being self-contained.

For any point x contained in the convex hull of a set of d points that are no further than δ apart from x, we show that the residual is bounded by $O(\|h\|_{\mathcal{H}}\,\delta^2)$, where $\|h\|_{\mathcal{H}}$ is the Hilbert Space norm of the associated function and that furthermore the posterior predictive variance is bounded by $O(\delta^2)$. Proposition 1 is our key approximation result. It plays a central role in the proof of our exponential regret bounds.

Lemma 5 (Hilbert Space Properties) *Given a set of points $x_{1:T} := \{x_1, \ldots, x_T\} \in \mathcal{D}$ and a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$ with kernel κ the following bounds hold:*

1. *Any $h \in \mathcal{H}$ is Lipschitz continuous with constant $\|h\|_{\mathcal{H}}\,L$, where $\|\cdot\|_{\mathcal{H}}$ is the Hilbert space norm and L satisfies the following:*
$$L^2 \leq \sup_{x\in\mathcal{D}} \partial_x \partial_{x'} \kappa(x, x')|_{x=x'} \tag{6}$$
and for $\kappa(x, x') = \widetilde{\kappa}(x - x')$ we have
$$L^2 \leq \partial_x^2 \widetilde{\kappa}(x)|_{x=0}.$$
2. *Any $h \in \mathcal{H}$ has its second derivative bounded by $\|h\|_{\mathcal{H}}\,Q$, where Q is any number satisfying*
$$Q^2 \leq \sup_{x\in\mathcal{D}} \partial_x^2 \partial_{x'}^2 \kappa(x, x')|_{x=x'} \tag{7}$$
and for $\kappa(x, x') = \widetilde{\kappa}(x - x')$ we have
$$Q^2 \leq \partial_x^4 \widetilde{\kappa}(x)|_{x=0}.$$
3. *The projection operator $P_{1:T}$ on the subspace $\operatorname*{span}_{t=1:T}\{\kappa(x_t, \cdot)\} \subseteq \mathcal{H}$ is given by*
$$P_{1:T} h := \mathbf{k}^\top(\cdot)\mathbf{K}^{-1} \langle \mathbf{k}(\cdot), h\rangle \tag{8}$$
where $\mathbf{k}(\cdot) = \mathbf{k}_{1:T}(\cdot) := [\kappa(x_1, \cdot) \cdots \kappa(x_T, \cdot)]^\top$ and $\mathbf{K} := [\kappa(x_i, x_j)]_{i,j=1:T}$; moreover, we have that
$$\langle \mathbf{k}(\cdot), h\rangle := \begin{bmatrix} \langle \kappa(x_1, \cdot), h\rangle \\ \vdots \\ \langle \kappa(x_T, \cdot), h\rangle \end{bmatrix} = \begin{bmatrix} h(x_1) \\ \vdots \\ h(x_T) \end{bmatrix}.$$
Here $P_{1:T}P_{1:T} = P_{1:T}$ and $\|P_{1:T}\| \leq 1$ and $\|\mathbf{1} - P_{1:T}\| \leq 1$.
4. *Given sets $x_{1:T} \subseteq x_{1:T'}$ it follows that $\|P_{1:T}h\|_{\mathcal{H}} \leq \|P_{1:T'}h\|_{\mathcal{H}} \leq \|h\|_{\mathcal{H}}$.*
5. *Given tuples (x_i, h_i) with $h_i = h(x_i)$, the minimum norm interpolation $\bar{h}$ with $\bar{h}(x_i) = h(x_i)$ is given by $\bar{h} = P_{1:T}h$. Consequently its residual $g := (\mathbf{1} - P_{1:T})h$ satisfies $g(x_i) = 0$ for all $x_i \in x_{1:T}$.*

Proof We prove the claims in sequence.

1. This follows from Corollary 4.36 in (Steinwart & Christmann, 2008), with $|\alpha| = 1$.
2. Same as above, just with $|\alpha| = 2$.
3. For any operator V with full column rank the projection on the image of V is given by $V(V^\top V)^{-1}V^\top$. The operator V in the above case is given by the stacked vector of evaluation functionals $k(x_1, \cdot), \ldots, k(x_n, \cdot)$. This provides us with P_X. The remaining claims are standard linear algebra.

4. Projection operators satisfy $\|P_{1:T}\| \leq 1$. This proves the second claim. The first claim can be seen from the fact that projecting on a subspace can only have a smaller norm than the superspace projection.

5. We first show that the projection is an interpolation. This follows from

$$\bar{h}(x_i) = P_{1:T}h(x_i) = \langle P_{1:T}h, \kappa(x_i, \cdot)\rangle = \langle h, P_{1:T}\kappa(x_i, \cdot)\rangle = \langle h, \kappa(x_i, \cdot)\rangle = h(x_i).$$

   Correspondingly $g(x_i) = h(x_i) - \bar{h}(x_i) = 0$ for all $x_i \in x_{1:T}$. By construction $P_{1:T}h$ uses h only in evaluations $h(x_i)$, hence for any two functions h, h' with $h(x_i) = h'(x_i)$ we have $P_{1:T}h = P_{1:T}h'$. Since $\|P_{1:T}\| \leq 1$ it follows that $\|P_{1:T}h\| \leq \|h\|_{\mathcal{H}}$. Hence there is no interpolation with norm smaller than $\|P_{1:T}h\|$.

■

Lemma 6 (GP Variance) *Under the assumptions of Lemma 5 it follows that*

$$|h(x) - P_{1:T}h(x)| \leq \|h\|_{\mathcal{H}}\, \sigma_T(x), \tag{9}$$

where $\sigma_T^2(x) = \kappa(x,x) - \mathbf{k}_{1:T}^\top(x)\mathbf{K}^{-1}\mathbf{k}_{1:T}(x)$ and this bound is tight. Moreover, $\sigma_T^2(x)$ is the posterior predictive variance of a Gaussian process with the same kernel.

Proof To see the bound we again use the Cauchy-Schwartz inequality

$$\begin{aligned}
|h(x) - P_{1:T}h(x)| &= |(\mathbf{1} - P_{1:T})h(x)| \\
&= |\langle(\mathbf{1} - P_{1:T})h, \kappa(x, \cdot)\rangle_{\mathcal{H}}| \quad \text{(by the defining property of } \langle\,,\rangle_{\mathcal{H}}, \\
&\qquad\qquad\qquad\qquad\qquad\qquad \text{cf. (Steinwart \& Christmann, 2008), Def. 4.18)} \\
&= |\langle h, (\mathbf{1} - P_{1:T})\kappa(x, \cdot)\rangle_{\mathcal{H}}| \quad \text{(since } \mathbf{1} - P_{1:T} \text{ is an orthogonal projection and so self-adjoint)} \\
&\leq \|h\|_{\mathcal{H}}\, \|(\mathbf{1} - P_{1:T})\kappa(x, \cdot)\| \quad \text{(by Cauchy-Schwarz)}
\end{aligned}$$

This inequality is clearly tight for $h = (\mathbf{1} - P_{1:T})\kappa(x, \cdot)$ by the nature of dual norms. Next note that

$$\begin{aligned}
\|(\mathbf{1} - P_{1:T})\kappa(x, \cdot)\|^2 &= \langle(\mathbf{1} - P_{1:T})\kappa(x, \cdot), (\mathbf{1} - P_{1:T})\kappa(x, \cdot)\rangle = \langle\kappa(x, \cdot), (\mathbf{1} - P_{1:T})\kappa(x, \cdot)\rangle \\
&= \kappa(x, x) - \langle\kappa(x, \cdot), P_{1:T}\kappa(x, \cdot)\rangle = \sigma_T^2(x).
\end{aligned}$$

The second equality follows from the fact that $\mathbf{1} - P_{1:T}$ is idempotent. The last equality follows from the definition of $P_{1:T}$. The fact that $\sigma_T^2(x)$ is the posterior predictive variance of a Gaussian Process regression estimate is well known in the literature and follows, e.g. from the matrix inversion lemma. ■

Lemma 7 (Approximation Guarantees) *We denote by $x_{1:T} \subseteq \mathcal{D}$ a set of locations and assume that $g(x_i) = 0$ for all $x_i \in x_{1:T}$.*

1. *Assume that g is Lipschitz continuous with bound L. Then $g(x) \leq Ld(x, x_{1:T})$, where $d(x, x_{1:T})$ is the minimum distance $\|x - x_i\|$ between x and any $x_i \in x_{1:T}$.*

2. *Assume that g has its second derivative bounded by Q'. Moreover, assume that x is contained inside the convex hull of $x_{1:T}$ such that the smallest such convex hull has a maximum pairwise distance between vertices of d. Then we have $g(x) \leq \frac{1}{4}Q'd^2$.*

Proof The first claim is an immediate consequence of the Lipschitz property of g. To see the second claim we need to establish a number of issues: without loss of generality assume that the maximum within the convex hull containing x is attained at x (and that the maximum rather than the minimum denotes the maximum deviation from 0).

The maximum distance of x to one of its vertices is bounded by $\delta/\sqrt{2}$. This is established by considering the minimum enclosing ball and realizing that the maximum distance is achieved for the regular polyhedron.

To see the maximum deviation from 0 we exploit the fact that $\partial_x g(x) = 0$ by the assumption of x being the maximum (we need not consider cases where x is on a facet of the polyhedral set since in this case we could easily reduce the dimensionality). In this case the largest deviation between $g(x)$ and $g(x_i)$ is obtained by making g a quadratic function $g(x') = \frac{Q'}{2} \|x' - x\|^2$. At distance $\frac{\delta}{\sqrt{2}}$ the function value is bounded by $\frac{\delta^2 Q'}{4}$. Since the latter bounds the maximum deviation it does bound it for g in particular. This proves the claim. ■

Proof [Proposition 1] Let $\mathcal{H}$ be the RKHS corresponding to κ and $h \in \mathcal{H}$ an arbitrary element, with $g := (\mathbf{1} - P_{1:T})h$ the residual defined in Lemma 5.5. By Lemma 5.3, we know that $\|\mathbf{1} - P_{1:T}\| \leq 1$ and so we have

$$\|g\|_{\mathcal{H}} \leq \|\mathbf{1} - P_{1:T}\| \, \|h\|_{\mathcal{H}} \leq \|h\|_{\mathcal{H}} \tag{10}$$

Moreover, by Lemma 5.2, we know that the second derivative of g is bounded by $\|g\|_{\mathcal{H}} Q$, and since by Lemma 5.5 we know that g vanishes at each x_i, we can use Lemma 7.2 and the inequality given by inequality (10) to conclude that

$$\begin{aligned} |h(x) - P_{1:T}h(x)| &:= |g(x)| \\ &\leq \frac{\|g\|_{\mathcal{H}} Q\delta^2}{4} \text{ by Lemma 7.2} \\ &\leq \frac{\|h\|_{\mathcal{H}} Q\delta^2}{4} \text{ by inequality (10)} \end{aligned}$$

and so for all $x \in \mathcal{D}$ we have

$$|h(x) - P_{1:T}h(x)| \leq \frac{Q\delta^2}{4} \|h\|_{\mathcal{H}} \tag{11}$$

On the other hand, by Lemma 6, we know that for all $x \in \mathcal{D}$ we have the following tight bound:

$$|h(x) - P_{1:T}h(x)| \leq \sigma_T(x) \|h\|_{\mathcal{H}} . \tag{12}$$

Now, given the fact that both inequalities (11) and (12) are bounding the same quantity and that the latter is a tight estimate, we necessarily have that

$$\sigma_T(x) \|h\|_{\mathcal{H}} \leq \frac{Q\delta^2}{4} \|h\|_{\mathcal{H}} .$$

Canceling $\|h\|_{\mathcal{H}}$ gives the desired result.

■

5.2. Finiteness of Regret

We begin with two lemmas from (Srinivas et al., 2010):

Lemma 8 (Lemma 5.1 of (Srinivas et al., 2010)) *Given any finite set $\mathcal{L}$, any sequence of points $\{x_1, x_2, \ldots\} \subseteq \mathcal{L}$ and $f : \mathcal{L} \to \mathbb{R}$ a sample from $\mathrm{GP}(0, \kappa(\cdot,\cdot))$, for all $\alpha \in (0,1)$, we have*

$$P\left\{\forall x \in \mathcal{L}, t \geq 1 : \ |f(x) - \mu_{t-1}(x)| \leq \sqrt{\beta_t}\sigma_{t-1}(x)\right\} \geq 1 - \alpha,$$

where $\beta_t = 2\ln\left(\frac{|\mathcal{L}|\pi_t}{\alpha}\right)$ and $\{\pi_t\}$ is any positive sequence satisfying $\sum_t \frac{1}{\pi_t} = 1$. Here $|\mathcal{L}|$ denotes the number of elements in $\mathcal{L}$.

Lemma 9 (Lemma 5.2 in (Srinivas et al., 2010)) *Let $\mathcal{L}$ a non-empty finite set and $f : \mathcal{L} \to \mathbb{R}$ an arbitrary function. Also assume that there exist functions $\mu, \sigma : \mathcal{L} \to \mathbb{R}$ and a constant $\sqrt{\beta}$, such that*

$$|f(x) - \mu(x)| \leq \sqrt{\beta}\sigma \quad \forall\, x \in \mathcal{L}. \tag{13}$$

Then, we have

$$r(x) \leq 2\sqrt{\beta}\sigma(x) \leq 2\sqrt{\beta} \max_{\mathcal{L}} \sigma.$$

Definition 10 (Covering Number) *Denote by $\mathcal{B}$ a Banach space with norm $\|\cdot\|$. Furthermore denote by $B \subseteq \mathcal{B}$ a set in this space. Then the covering number $n_\epsilon(B, \mathcal{B})$ is defined as the minimum number of ϵ balls with respect to the Banach space norm that are required to cover B entirely.*

Proof [Theorem 2] The proof consists of the following steps:

- **Global:** We first show that after a finite number of steps the algorithm zooms in on the neighbourhood $B(x_M, \rho_0)$. This is done by first showing that ϵ can be chosen small enough to squeeze the set $f^{-1}((f_M - \epsilon, f_M])$ into any arbitrarily small neighbourhood of x_M and that as the function is sampled more and more densely, the UBC-LCB envelope around f becomes arbitrarily tight, hence eventually fitting the relevant set inside a small neighbourhood of x_M. Please refer to Figure 4 for a graphical depiction of this process.

  G$_I$: Since $\mathcal{D}$ is compact and f is continuous and has a unique maximum, for every $\rho > 0$, we can find an $\epsilon = \epsilon(\rho) > 0$ such that

  $$f^{-1}\left((f_M - \epsilon, f_M]\right) \subseteq B(x_M, \rho),$$

  where $f_M = \max f$.
  To see this, suppose on the contrary that there exists a radius $\rho > 0$ such that for all $\epsilon > 0$ we have

  $$f^{-1}\left((f_M - \epsilon, f_M]\right) \nsubseteq B(x_M, \rho)$$

  which means that there exists a point $x \in \mathcal{D}$ such that $f(x_M) - f(x) < \epsilon$ but $\|x - x_M\| > \rho$. Now, for each $i \in \mathbb{N}$, pick a point $x^i \in f^{-1}\left((f_M - \frac{1}{i}, f_M]\right) \setminus B(x_M, \rho)$: this gives us a sequence of points $\{x^i\}$ in $\mathcal{D}$, which by the compactness of $\mathcal{D}$ has a convergent subsequence $\{x^{i_k}\}$, whose limit we will denote by x^*. From the continuity of f and the fact that $f(x_M) - f(x^i) < \frac{1}{i}$, we can conclude that $f(x_M) - f(x^*) = 0$, which contradicts our assumption that f has a unique global maximum since we necessarily have $x^* \notin B(x_M, \rho)$.

  G$_{II}$: Define $\epsilon^* := \dfrac{\epsilon(\rho_0)}{4}$, with ρ_0 as in Condition (†) of the statement of Theorem 2.

  G$_{III}$: For each T, define the "relevant set" $\mathcal{R}_T \subseteq \mathcal{D}$ as follows:

  $$\mathcal{R}_T = \left\{ x \in \mathcal{D} \middle| \mu_T(x) + \sqrt{\beta_T}\sigma_T(x) > \sup_{\mathcal{R}} \mu_T(x) - \sqrt{\beta_T}\sigma_T(x) \right\}.$$

  G$_{IV}$: Choose $\beta_T = b\ln(T)$, with b chosen large enough to satisfy the conditions of Lemma 8. Then, it is possible to sample f densely enough so that

  $$\sqrt{\beta_T} \max_{x \in \mathcal{D}} \sigma_T(x) < \epsilon^*, \tag{14}$$

  so that $\mathcal{R}_T \subseteq B(x_M, \rho_0)$. This is because as $\mathcal{D}$ is sampled more and more densely we have $\sigma = O(\delta^2)$, where δ is the distance between the points of the grid, and $\beta = O\left(\ln \frac{1}{\delta^d}\right) = O\left(-\ln \delta\right)$ and so $\sqrt{\beta}\sigma \to 0$ as $\delta \to 0$, and so there exists a δ_0 small enough so that a lattice of resolution δ_0 would give us the bound given in inequality (14). The end point of this process is depicted in Figure 4, where the relevant set $\mathcal{R}_T$ lies inside the non-shaded region: the reason for this inclusion and "thickness" $4\epsilon^*$ is described below, in Step L$_1$ of the proof: cf. Equation (15).

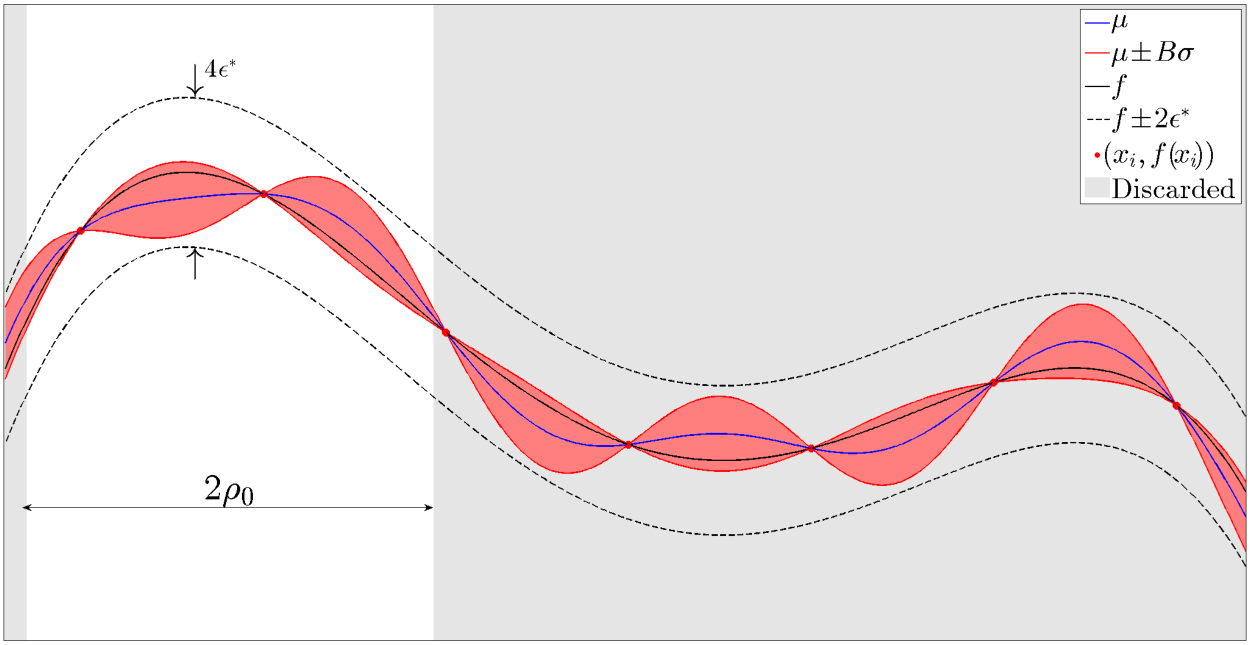


Figure 4. The elimination of other smaller peaks.

- **Local:** Once the algorithm has localized attention to a neighbourhood of x_M, then we can show that the regret decreases exponentially; to do so, we will proceed by sampling the relevant set twice as densely and shrinking the relevant set, and repeating these two steps. The claim is that in each iteration, the maximum regret goes down exponentially and the number of the new points that are sampled in each **refining** iteration is asymptotically constant. To prove this, we will write down the equations governing the behaviour of the number of sampled points and σ. We will adopt the following notation to carry out this task:

  - δ_ℓ - the resolution of the lattice of sampled points at the end of the $(\ell+1)^{th}$ refining iteration inside $\mathcal{R}_{\ell+1}$ (defined below).
  - $\epsilon_\ell = \sup_{x\in\mathcal{R}_\ell} \sigma_{N_\ell}(x)$ at the end of the ℓ^{th} iteration. Note that $\epsilon_\ell \propto \delta_\ell^2$. Also, note that $\epsilon_0 \leq \epsilon^*$ by the choice of δ_0.
  - N_ℓ - number of points that have been sampled by the end of the ℓ^{th} iteration.
  - $\Delta N_\ell = N_{\ell+1} - N_\ell$.
  - $\mathcal{R}_\ell$ - the relevant set at the beginning of the ℓ^{th} iteration. Note that $\mathcal{R}_1 \subseteq B(x_M, \rho_0)$.
  - $\rho_\ell = \dfrac{\text{diam}(\mathcal{R}_\ell)}{2}$. Note that $\rho_1 < \rho_0$.

L$_1$: We have the following chain of inequalities:

$$
\begin{aligned}
N_1 &\leq N_0 + W n_{\delta_0}\left(\mathcal{R}_0, (\mathbb{R}^d, \|\cdot\|_2)\right) && \text{where } n_{\delta_0}\left(\mathcal{R}_0, (\mathbb{R}^d, \|\cdot\|_2)\right) \text{ is the } \delta_0\text{-covering number} \\
& && \text{as defined in Definition 10} \\
&\leq N_0 + W\mathcal{N}(\rho_0, \delta_0) && \text{where } \mathcal{N}(\rho, \delta) := n_\delta\left(B(0,\rho), (\mathbb{R}^d, \|\cdot\|_2)\right) \\
&\leq N_0 + W\mathcal{N}\left(\sqrt{\frac{4\epsilon_0\sqrt{\beta_{N_0}}}{c_2}}, \delta_0\right) \\
&\leq N_0 + W\mathcal{N}\left(\sqrt{\frac{4\epsilon_0\sqrt{b\ln N_0}}{c_2}}, \delta_0\right) \\
&= N_0 + W\mathcal{N}\left(c\sqrt{\epsilon_0}\sqrt[4]{\ln N_0}, \delta_0\right) && \text{where } c := \sqrt{\frac{4\sqrt{b}}{c_2}}
\end{aligned}
$$

In the first line of the above chain of inequalities, the factor W multiplying the last term is any number greater than one that gives an upper bound on the algorithm's wastefulness when it comes to producing a δ-covering of any set $\mathcal{S} \subseteq \mathbb{R}^d$ as compared to the minimum number of points necessary, i.e. $n_\delta\left(\mathcal{S}, (\mathbb{R}^d, \|\cdot\|_2)\right)$.

The expression $\sqrt{\frac{4\epsilon_0\sqrt{\beta_{N_0}}}{c_2}}$ comes about as follows: using the notations $B = \sqrt{\beta_{N_0}}$ and $\sigma = \sigma_{N_0}$ we know by Lemma 8 that f and μ are intertwined with each other in the sense that both of the following chains of inequality hold:

$$
\begin{array}{ccccc}
\mu - B\sigma & \leq & f & \leq & \mu + B\sigma \\
f - B\sigma & \leq & \mu & \leq & f + B\sigma,
\end{array}
$$

which, combined together, give us the following chain of inequalities

$$
f - 2B\sigma \quad \leq \quad \mu - B\sigma \quad \leq \quad f \quad \leq \quad \mu + B\sigma \quad \leq \quad f + 2B\sigma. \tag{15}
$$

Since, we also know that $\sigma(x) \leq \epsilon_0$ for all $x \in \mathcal{R}_0$, we can conclude that

$$
f - 2B\epsilon_0 \quad \leq \quad \mu - B\sigma \quad \leq \quad \mu + B\sigma \quad \leq \quad f + 2B\epsilon_0.
$$

Moreover, if condition (†) holds, we know that in $\mathcal{R}_0$, the function f satisfies $-c_1\boldsymbol{r}^2 < f(x) - f(x_M) < -c_2\boldsymbol{r}^2$, where $\boldsymbol{r} = \boldsymbol{r}(x) := \|x - x_M\|$, so we get that

$$
f(x_M) - c_1\boldsymbol{r}^2 - 2B\epsilon_0 \quad \leq \quad \mu - B\sigma \quad \leq \quad \mu + B\sigma \quad \leq \quad f(x_M) - c_2\boldsymbol{r}^2 + 2B\epsilon_0.
$$

Now, recall that $\mathcal{R}_0$ is defined to consist of points x where $\mu(x) + B\sigma(x) \geq \sup_{\mathcal{D}} \mu(x) - B\sigma(x)$, but given the fact that we have the above outer envelope for $\mu \pm B\sigma$, we can conclude that

$$
\begin{aligned}
\mathcal{R}_0 &\subseteq \left\{x \,\middle|\, f(x_M) - c_2\boldsymbol{r}(x)^2 + 2B\epsilon_0 \ \geq \ \max_{x \in \mathcal{D}} f(x_M) - c_1\boldsymbol{r}(x)^2 - 2B\epsilon_0\right\} \\
&= \left\{x \,\middle|\, f(x_M) - c_2\boldsymbol{r}(x)^2 + 2B\epsilon_0 \ \geq \ f(x_M) - 2B\epsilon_0\right\} \\
&= \left\{x \,\middle|\, -c_2\boldsymbol{r}(x)^2 + 2B\epsilon_0 \ \geq \ -2B\epsilon_0\right\} \\
&= \left\{x \,\middle|\, c_2\boldsymbol{r}(x)^2 \ \leq \ 4B\epsilon_0\right\} \\
&= \left\{x \,\middle|\, \boldsymbol{r}(x) \ \leq \ \sqrt{\frac{4B\epsilon_0}{c_2}}\right\}
\end{aligned}
$$

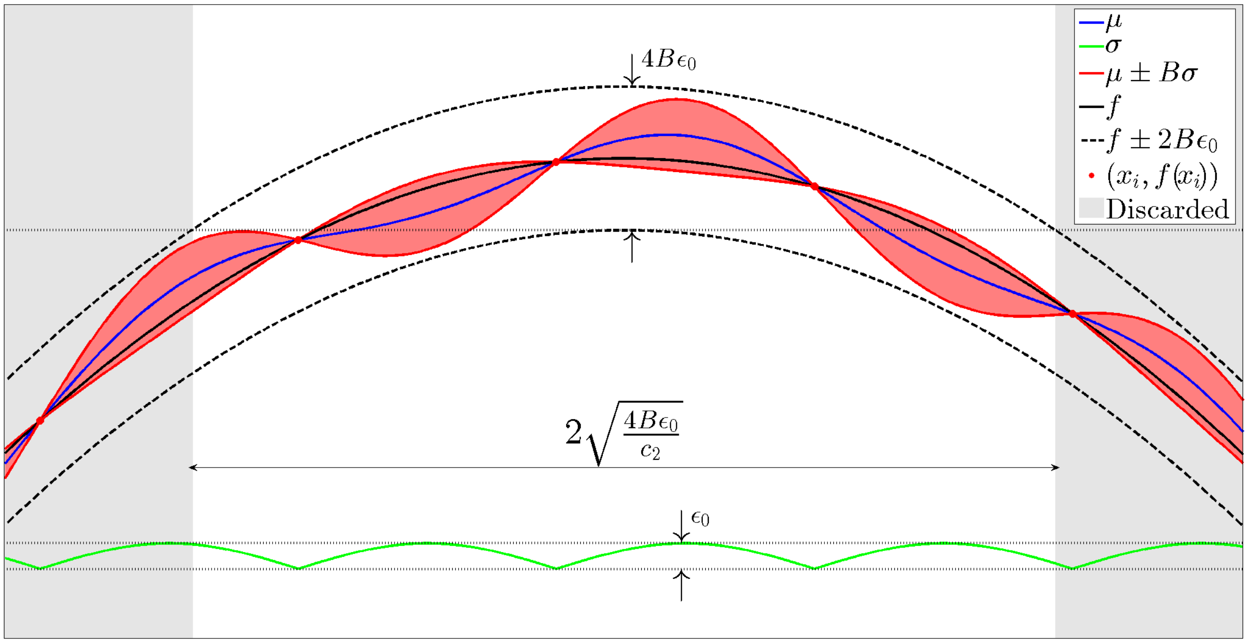


Figure 5. The shrinking of the relevant set $\mathcal{R}_\ell$. Here, $B = \sqrt{\beta_{N_0}}$

Now, if, on the other hand, f satisfies condition (‡), then by the smoothness assumptions in (‡), we know that $\nabla f(x_M)$ is perpendicular to $\partial\mathcal{D}$ at x_M and so there exist positive numbers c_1 and c_2 such that in a neighbourhood of x_M we have

$$-c_1\boldsymbol{r} \quad \leq \quad f - f(x_M) \quad \leq \quad -c_2\boldsymbol{r}^2.$$

Note that in the argument above in the case of (†), the precise form of the lower bound on f was irrelevant, since all we are interested in is its maximum. So, the same argument goes through again. This is depicted in Figure 5, where $B := \sqrt{\beta_{N_0}} = \sqrt{b \ln N_0}$.

$\mathrm{L}_{\ell+1}$: Now, let us suppose that we are the end of the ℓ^{th} iteration. We have

$$\begin{aligned}
N_{\ell+1} &\leq N_\ell + W\mathcal{N}(\rho_\ell, \delta_\ell) \\
&= N_\ell + W\mathcal{N}\left(c\sqrt{\epsilon_\ell}\sqrt[4]{\ln N_\ell}, \delta_\ell\right) \\
&\leq N_\ell + W\mathcal{N}\left(c\sqrt{\frac{\epsilon_0}{4^\ell}}\sqrt[4]{\ln N_\ell}, \frac{\delta_0}{2^\ell}\right) && \text{by Proposition 1} \\
&= N_\ell + W\mathcal{N}\left(c\sqrt{\epsilon_0}\sqrt[4]{\ln N_\ell}, \delta_0\right) && \text{since } \mathcal{N}(2\rho, 2\delta) = \mathcal{N}(\rho, \delta) \text{ for any } \rho \text{ and } \delta \\
&:= N_\ell + Wn_{\delta_0}\left(B(0, c\sqrt{\epsilon_0}\sqrt[4]{\ln N_\ell}), (\mathbb{R}^d, \|\cdot\|_2)\right) && \text{by the definition of } \mathcal{N} \\
&\leq N_\ell + Wn_{\delta_0}\left(\left[-c\sqrt{\epsilon_0}\sqrt[4]{\ln N_\ell}, c\sqrt{\epsilon_0}\sqrt[4]{\ln N_\ell}\right]^d, (\mathbb{R}^d, \|\cdot\|_2)\right) \\
&\leq N_\ell + W\left(\frac{2c\sqrt{\epsilon_0}\sqrt[4]{\ln N_\ell}}{\delta_0}\right)^d && \text{since a regular lattice of resolution } \delta_0 \text{ gives a } \delta_0\text{-covering} \\
&\leq N_\ell + C(\ln N_\ell)^{\frac{d}{4}} && \text{where } C = W\left(\frac{2c\sqrt{\epsilon_0}}{\delta_0}\right)^d
\end{aligned}$$

So, the number of samples needed by the branch and bound algorithm is governed by the difference inequation

$$\Delta N_\ell \leq C(\ln N_\ell)^{\frac{d}{4}}. \tag{16}$$

To study the solutions of this difference equation, we consider the corresponding differential equation:

$$\frac{dN}{d\ell} = C(\ln N)^{\frac{d}{4}}. \tag{17}$$

Since this equation is separable, we can write

$$\frac{dN}{(\ln N)^{\frac{d}{4}}} = Cd\ell.$$

Now, letting $\ell = L$ be a given number of iterations in the algorithm and $N(L)$ the corresponding number of sampled points, we can integrate both sides of the above equation to get

$$\int_{N(0)}^{N(L)} \frac{dN}{(\ln N)^{\frac{d}{4}}} = \int_0^L Cd\ell = CL.$$

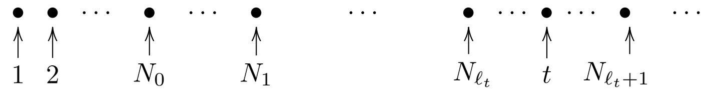
Given the fact that the integral on the left can't be solved analytically, we will use the lower bound

$$\frac{N(L)-N(0)}{(\ln N(L))^{\frac{d}{4}}} \leq \int_{N(0)}^{N(L)} \frac{dN}{(\ln N)^{\frac{d}{4}}}$$

to get

$$\frac{N(L)-N(0)}{C(\ln N(L))^{\frac{d}{4}}} \leq L \tag{18}$$

Given a time t, we will denote by ℓ_t the largest non-negative integer such that $N_{\ell_t} < t$ or 0 if no such number exists. We illustrate this somewhat obtuse definition with the following example:



Now, by Lemma 9, for all $t >> N_0$ we have

$$\begin{aligned}
r_t &\leq 2\sqrt{\beta_t}\max_{\mathcal{R}_{\ell_t}} \sigma_t \leq 2\sqrt{b\ln t}\epsilon_{\ell_t} \leq \frac{2\epsilon_0\sqrt{b\ln t}}{4^{\ell_t}} \leq \frac{8\epsilon_0\sqrt{b\ln t}}{4^{\ell_t+1}} \\
&\leq 8\epsilon_0\sqrt{b\ln t}\left(\frac{1}{4}\right)^{\frac{N_{\ell_t+1}-N_0}{C\left(\ln N_{\ell_t+1}\right)^{d/4}}} \qquad \text{by Equation 18} \\
&\leq 8\epsilon_0\sqrt{b\ln t}\left(\frac{1}{4}\right)^{\frac{DN_{\ell_t+1}}{\left(\ln N_{\ell_t+1}\right)^{d/4}}} \qquad \text{for some } D > 0 \text{ since } N_{\ell_t+1} > N_0 \\
&\leq 8\epsilon_0\sqrt{b\ln t}\left(\frac{1}{4}\right)^{\frac{Dt}{(\ln t)^{d/4}}} \qquad \text{for } t \text{ satisfying } \ln t > \frac{d}{4} \text{ (see } \star \text{ below) since } t \leq N_{\ell_t+1} \\
&\leq 8\epsilon_0\sqrt{b}e^{-\frac{Et}{(\ln t)^{d/4}}+\frac{\ln\ln t}{2}} \\
&\leq 8\epsilon_0\sqrt{b}e^{-\frac{Et}{(\ln t)^{d/4}}+\frac{Et}{2(\ln t)^{d/4}}} \qquad \text{for large enough } t \\
&= Ae^{-\frac{\tau t}{(\ln t)^{d/4}}} \qquad \text{for } A = 8\epsilon_0\sqrt{b} \text{ and } \tau = E/2.
\end{aligned}$$

$\star$ The reason for the specific criterion $\ln t > \frac{d}{4}$ is that the function $\frac{x}{(\ln x)^{d/4}}$ is increasing when this condition is satisfied, and so decreasing x from $N_{\ell_t}+1$ to t decreases its value, increasing the overall expression $\left(\frac{1}{4}\right)^{\frac{x}{(\ln x)^{d/4}}}$. To see that $\frac{x}{(\ln x)^{d/4}}$ becomes increasing when $\ln x > \frac{d}{4}$, we simply need to calculate its derivative:

$$\begin{aligned}
\frac{d}{dx}\frac{x}{(\ln x)^{d/4}} &= \frac{1}{(\ln x)^{d/4}} - \frac{d}{4}\frac{x}{x(\ln x)^{d/4+1}} \\
&= \frac{\ln x - \frac{d}{4}}{(\ln x)^{d/4}}.
\end{aligned}$$

Moreover, since $N_{\ell_t+1} \geq t$, if the derivative of $\frac{x}{(\ln x)^{d/4}}$ is positive at t, it is also positive between t and N_{ℓ_t+1} and so the function is indeed increasing in that interval.

■