Multi-fidelity Bayesian Optimisation with Continuous Approximations

Kirthevasan Kandasamy [1] **Gautam Dasarathy** [2] **Jeff Schneider** [1] **Barnabás Póczos** [1]

Abstract

Bandit methods for black-box optimisation, such as Bayesian optimisation, are used in a variety of applications including hyper-parameter tuning and experiment design. Recently, *multi-fidelity* methods have garnered considerable attention since function evaluations have become increasingly expensive in such applications. Multi-fidelity methods use cheap approximations to the function of interest to speed up the overall optimisation process. However, most multi-fidelity methods assume only a finite number of approximations. On the other hand, in many practical applications, a continuous spectrum of approximations might be available. For instance, when tuning an expensive neural network, one might choose to approximate the cross validation performance using less data N and/or few training iterations T. Here, the approximations are best viewed as arising out of a continuous two dimensional space (N, T). In this work, we develop a Bayesian optimisation method, BOCA, for this setting. We characterise its theoretical properties and show that it achieves better regret than than strategies which ignore the approximations. BOCA outperforms several other baselines in synthetic and real experiments.

1. Introduction

Many tasks in scientific and engineering applications can be framed as *bandit optimisation* problems, where we need to sequentially evaluate a noisy black-box function $f : \mathcal{X} \to \mathbb{R}$ with the goal of finding its optimum. Some applications include hyper-parameter tuning in machine learning (Hutter et al., 2011; Snoek et al., 2012), optimal policy search (Lizotte et al., 2007; Martinez-Cantin et al., 2007) and scientific experiments (Gonzalez et al., 2014; Parkinson et al., 2006). Typically, in such applications, each function evaluation is expensive, and conventionally, the bandit literature has focused on developing methods for finding the optimum while keeping the number of evaluations to f at a minimum.

However, with increasingly expensive function evaluations, conventional methods have become infeasible as a significant cost needs to be expended before we can learn anything about f. As a result, *multi-fidelity* optimisation methods have recently gained attention (Cutler et al., 2014; Kandasamy et al., 2016a; Li et al., 2016). As the name suggests, these methods assume that we have access to lower fidelity approximations to f which can be evaluated instead of f. The lower the fidelity, the cheaper the evaluation, but it provides less accurate information about f. For example, when optimising the configuration of an expensive real world robot, its performance can be approximated using cheaper computer simulations. The goal is to use the cheap approximations to guide search for the optimum of f, and reduce the overall cost of optimisation. However, most multi-fidelity work assume only a finite number of approximations. In this paper, we study multi-fidelity optimisation when there is access to a continuous spectrum of approximations.

To motivate this set up, consider tuning a classification algorithm over a space of hyper-parameters $\mathcal{X}$ by maximising a validation set accuracy. The algorithm is to be trained using $N_\bullet$ data points via an iterative algorithm for $T_\bullet$ iterations. However, we wish to use fewer training points $N < N_\bullet$ and/or fewer iterations $T < T_\bullet$ to approximate the validation accuracy. We can view validation accuracy as a function $g : [1, N_\bullet] \times [1, T_\bullet] \times \mathcal{X} \to \mathbb{R}$ where evaluating $g(N, T, x)$ requires training the algorithm with N points for T iterations with the hyper-parameters x. If the training complexity of the algorithm is quadratic in data size and linear in the number of iterations, then the cost of this evaluation is $\lambda(N, T) = \mathcal{O}(N^2 T)$. Our goal is to find the optimum when $N = N_\bullet$, and $T = T_\bullet$, i.e. we wish to maximise $f(x) = g(N_\bullet, T_\bullet, x)$.

In this setting, while N, T are technically discrete choices, they are more naturally viewed as coming from a continuous 2 dimensional *fidelity space*, $[1, N_\bullet] \times [1, T_\bullet]$. One might hope that cheaper queries to $g(N, T, \cdot)$ with N, T less than $N_\bullet, T_\bullet$ can be used to learn about $g(N_\bullet, T_\bullet, \cdot)$ and consequently optimise it using less overall cost. Indeed, this

*Equal contribution [1]Carnegie Mellon University, Pittsburgh PA, USA [2]Rice University, Houston TX, USA. Correspondence to: Kirthevasan Kandasamy <kandasamy@cmu.edu>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

is the case with many machine learning algorithms where cross validation performance tends to vary smoothly with data set size and number of iterations. Therefore, one may use cheap low fidelity experiments with small (N, T) to discard bad hyper-parameters and deploy expensive high fidelity experiments with large (N, T) only in a small but promising region. The main theoretical result of this paper (Theorem 1) shows that our proposed algorithm, BOCA, exhibits precisely this behaviour.

Continuous approximations also arise in simulation studies: where simulations can be carried out at varying levels of granularity, on-line advertising: where an ad can be controlled by continuous parameters such as display time or target audience, and several other experiment design tasks. In fact, in many multi-fidelity papers, the finite approximations were obtained by discretising a continuous space (Huang et al., 2006; Kandasamy et al., 2016a). Here, we study a Bayesian optimisation technique that is directly designed for continuous fidelity spaces and is potentially applicable to more general spaces. Our **main contributions** are,

1. A novel setting and model for multi-fidelity optimisation with continuous approximations using Gaussian process (GP) assumptions. We develop a novel algorithm, BOCA, for this setting.
2. A theoretical analysis characterising the behaviour and regret bound for BOCA.
3. An empirical study which demonstrates that BOCA outperforms alternatives, both multi-fidelity and otherwise, on a series of synthetic problems and real examples in hyper-parameter tuning and astrophysics.

Related Work

Bayesian optimisation (BO), refers to a suite of techniques for bandit optimisation which use a prior belief distribution for f. While there are several techniques for BO (de Freitas et al., 2012; Hernández-Lobato et al., 2014; Jones et al., 1998; Mockus, 1994; Thompson, 1933), our work will build on the Gaussian process upper confidence bound (GP-UCB) algorithm of Srinivas et al. (2010). GP-UCB models f as a GP and uses upper confidence bound (UCB) (Auer, 2003) techniques to determine the next point for evaluation.

BO techniques have been used in developing multi-fidelity optimisation methods in various applications such as hyper-parameter tuning and industrial design (Forrester et al., 2007; Huang et al., 2006; Klein et al., 2015; Lam et al., 2015; Poloczek et al., 2016; Swersky et al., 2013). However, these methods are either problem specific and/or only use a finite number of fidelities. Further, none of them come with theoretical underpinnings. Recent work has studied multi-fidelity methods for specific problems such as hyper-parameter tuning, active learning and reinforcement learning (Agarwal et al., 2011; Cutler et al., 2014; Li et al., 2016; Sabharwal et al., 2015; Zhang & Chaudhuri, 2015). While some of the above tasks can be framed as optimisation problems, the methods themselves are specific to the problem considered. Our method is more general as it applies to any bandit optimisation task.

Perhaps the closest work to us is that of Kandasamy et al. (2016a;b;c) who developed MF-GP-UCB assuming a finite number of approximations to f. While this line of work was the first to provide theoretical guarantees for multi-fidelity optimisation, it has two important shortcomings. First, they make strong assumptions, particularly a uniform bound on the difference between the expensive function and an approximation. This does not allow for instances where an approximation might be good at certain regions but not at the other. In contrast, our probabilistic treatment between fidelities is is robust to such cases. Second, their model does not allow sharing information between fidelities; each approximation is treated independently. Not only is this wasteful as lower fidelities can provide useful information about higher fidelities, it also means that the algorithm might perform poorly if the fidelities are not designed properly. We demonstrate this with an experiment in Section 4. On the other hand, our model allows sharing information across the fidelity space in a natural way. In addition, we can also handle continuous approximations whereas their method is strictly for a finite number of approximations. That said, BOCA inherits a key intuition from MF-GP-UCB, which is to choose a fidelity only if we have sufficiently reduced the uncertainty at all lower fidelities. Besides this, there are considerable differences in the mechanics of the algorithm and proof techniques. As we proceed, we will draw further comparisons to Kandasamy et al. (2016a).

2. Preliminaries

2.1. Some Background Material

Gaussian processes: A GP over a space $\mathcal{X}$ is a random process from $\mathcal{X}$ to $\mathbb{R}$. GPs are typically used as a prior for functions in Bayesian nonparametrics. It is characterised by a mean function $\mu : \mathcal{X} \to \mathbb{R}$ and a covariance function (or kernel) $\kappa : \mathcal{X}^2 \to \mathbb{R}$. If $f \sim \mathcal{GP}(\mu, \kappa)$, then $f(x)$ is distributed normally $\mathcal{N}(\mu(x), \kappa(x, x))$ for all $x \in \mathcal{X}$. Suppose that we are given n observations $\mathcal{D}_n = \{(x_i, y_i)\}_{i=1}^n$ from this GP, where $x_i \in \mathcal{X}$, $y_i = f(x_i) + \epsilon_i \in \mathbb{R}$ and $\epsilon_i \sim \mathcal{N}(0, \eta^2)$. Then the posterior process $f|\mathcal{D}_n$ is also a GP with mean μ_n and covariance κ_n given by

$$\mu_n(x) = k^\top (K + \eta^2 I)^{-1} Y, \qquad (1)$$
$$\kappa_n(x, x') = \kappa(x, x') - k^\top (K + \eta^2 I)^{-1} k'.$$

Here $Y \in \mathbb{R}^n$ is a vector with $Y_i = y_i$, and $k, k' \in \mathbb{R}^n$ are such that $k_i = \kappa(x, x_i), k'_i = \kappa(x', x_i)$. The matrix $K \in \mathbb{R}^{n \times n}$ is given by $K_{i,j} = \kappa(x_i, x_j)$. We refer the reader to chapter 2 of Rasmussen & Williams (2006) for more on the basics of GPs and their use in regression.

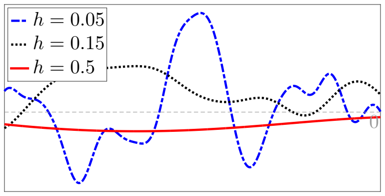


Figure 1. Samples drawn from a GP with 0 mean and SE kernel with bandwidths $h = 0.01, h = 0.15, 0.5$. Samples tend to be smoother across the domain for large bandwidths.

Radial kernels: The prior covariance functions of GPs are typically taken to be *radial kernels*; some examples are the squared exponential (SE) and Matérn kernels. Using a radial kernel means that the prior covariance can be written as $\kappa(x, x') = \kappa_0 \phi(\|x - x'\|)$ and depends only on the distance between x and x'. Here, the scale parameter κ_0 captures the magnitude f could deviate from μ. The function $\phi : \mathbb{R}_+ \to \mathbb{R}_+$ is a decreasing function with $\|\phi\|_\infty = \phi(0) = 1$. In this paper, we will use the SE kernel in a running example to convey the intuitions in our methods. For the SE kernel, $\phi(r) = \phi_h(r) = \exp(-r^2/(2h^2))$, where $h \in \mathbb{R}_+$, called the *bandwidth* of the kernel, controls the smoothness of the GP. When h is large, the samples drawn from the GP tend to be smoother as illustrated in Fig. 1. We will reference this observation frequently in the text.

GP-UCB: The Gaussian Process Upper Confidence Bound (GP-UCB) algorithm of Srinivas et al. (2010) is a method for bandit optimisation, which, like many other BO methods, models f as a sample from a Gaussian process. At time t, the next point x_t for evaluating f is chosen via the following procedure. First, we construct an upper confidence bound $\varphi_t(x) = \mu_{t-1}(x) + \beta_t^{1/2}\sigma_{t-1}(x)$ for the GP. μ_{t-1} is the posterior mean of the GP conditioned on the previous $t-1$ evaluations and σ_{t-1} is the posterior standard deviation. Following other UCB algorithms (Auer, 2003), the next point is chosen by maximising φ_t, i.e. $x_t = \mathrm{argmax}_{x \in \mathcal{X}} \varphi_t(x)$. The μ_{t-1} term encourages an *exploitative* strategy – in that we want to query regions where we already believe f is high – and σ_{t-1} encourages an *exploratory* strategy – in that we want to query where we are uncertain about f so that we do not miss regions which have not been queried yet. β_t, which is typically increasing with t, controls the trade-off between exploration and exploitation. We have provided a brief review of GP-UCB in Appendix A.1.

2.2. Problem Set Up

Our goal in bandit optimisation is to maximise a function $f : \mathcal{X} \to \mathbb{R}$, over a domain $\mathcal{X}$. When we evaluate f at $x \in \mathcal{X}$ we observe $y = f(x) + \epsilon$ where $\mathbb{E}[\epsilon] = 0$. Let $x_\star \in \mathrm{argmax}_{x\in\mathcal{X}} f(x)$ be a maximiser of f and $f_\star = f(x_\star)$ be the maximum value. An algorithm for bandit optimisation is a sequence of points $\{x_t\}_{t\geq 0}$, where, at time t, the algorithm chooses to evaluate f at x_t based on previous queries and

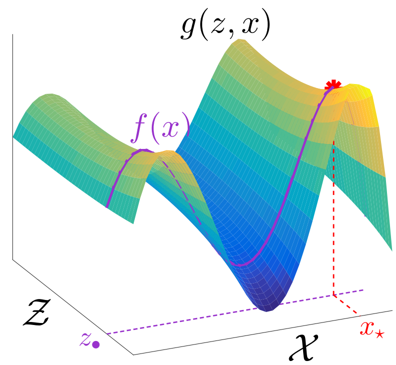


Figure 2. $g : \mathcal{Z} \times \mathcal{X} \to \mathbb{R}$ is a function defined on the product space of the fidelity space $\mathcal{Z}$ and domain $\mathcal{X}$. The purple line is $f(x) = g(z_\bullet, x)$. We wish to find the maximiser $x_\star \in \mathrm{argmax}_{x\in\mathcal{X}} f(x)$. The multi-fidelity framework is attractive when g is smooth across $\mathcal{Z}$ as illustrated in the figure.

observations $\{(x_i, y_i)\}_{i=1}^{t-1}$. After n queries to f, its goal is to achieve small *simple regret* S_n, as defined below.

$$S_n = \min_{t=1,\dots,n} f_\star - f(x_t). \tag{2}$$

Continuous Approximations: In this work, we will let f be a slice of a function g that lies in a larger space. Precisely, we will assume the existence of a fidelity space $\mathcal{Z}$ and a function $g : \mathcal{Z} \times \mathcal{X} \to \mathbb{R}$ defined on the product space of the fidelity space and domain. The function f which we wish to maximise is related to g via $f(\cdot) = g(z_\bullet, \cdot)$, where $z_\bullet \in \mathcal{Z}$. For instance, in the hyper-parameter tuning example from Section 1, $\mathcal{Z} = [1, N_\bullet] \times [1, T_\bullet]$ and $z_\bullet = [N_\bullet, T_\bullet]$. Our goal is to find a maximiser $x_\star \in \mathrm{argmax}_x f(x) = \mathrm{argmax}_x g(z_\bullet, x)$. We have illustrated this setup in Fig. 2. In the rest of the manuscript, the term "fidelities" will refer to points z in the fidelity space $\mathcal{Z}$.

The multi-fidelity framework is attractive when the following two conditions are true about the problem.

1. There exist fidelities $z \in \mathcal{Z}$ where evaluating g is cheaper than evaluating at $z_\bullet$. To this end, we will associate a *known* cost function $\lambda : \mathcal{Z} \to \mathbb{R}_+$. In the hyper-parameter tuning example, $\lambda(z) = \lambda(N, T) = \mathcal{O}(N^2 T)$. It is helpful to think of $z_\bullet$ as being the most expensive fidelity, i.e. maximiser of λ, and that $\lambda(z)$ decreases as we move away from $z_\bullet$. However, this notion is strictly not necessary for our algorithm or results.
2. The cheap $g(z, \cdot)$ evaluation gives us information about $g(z_\bullet, \cdot)$. This is true if g is smooth across the fidelity space as illustrated in Fig. 2. As we will describe shortly, this smoothness can be achieved by modelling g as a GP with an appropriate kernel for the fidelity space $\mathcal{Z}$.

In the above setup, a multi-fidelity algorithm is a sequence of query-fidelity pairs $\{(z_t, x_t)\}_{t\geq 0}$ where, at time t, the algorithm chooses $z_t \in \mathcal{Z}$ and $x_t \in \mathcal{X}$, and observes $y_t =$

$g(z_t, x_t) + \epsilon$ where $\mathbb{E}[\epsilon] = 0$. The choice of (z_t, x_t) can of course depend on the previous fidelity-query-observation triples $\{(z_i, x_i, y_i)\}_{i=1}^{t-1}$.

Multi-fidelity Simple Regret: We provide bounds on the simple regret $S(\Lambda)$ of a multi-fidelity optimisation method after it has spent capital Λ of a resource. Following Kandasamy et al. (2016a); Srinivas et al. (2010), we will aim to provide *any capital* bounds, meaning that an algorithm would be expected to do well for *all* values of (sufficiently large) Λ. Say we have made N queries to g within capital Λ, i.e. N is the *random* quantity such that $N = \max\{n \geq 1 : \sum_{t=1}^{n} \lambda(z_t) \leq \Lambda\}$. While the cheap evaluations at $z \neq z_\bullet$ are useful in guiding search for the optimum of $g(z_\bullet, \cdot)$, there is no reward for optimising a cheaper $g(z, \cdot)$. Accordingly, we define the simple regret after capital Λ as,

$$S(\Lambda) = \begin{cases} \min\limits_{\substack{t \in \{1,\dots,N\} \\ \text{s.t } z_t = z_\bullet}} f_\star - f(x_t) & \text{if we have queried at } z_\bullet, \\ +\infty & \text{otherwise.} \end{cases}$$

This definition reduces to the single fidelity definition (2) when we only query g at $z_\bullet$. It is also similar to the definition in Kandasamy et al. (2016a), but unlike them, we do not impose additional boundedness constraints on f or g.

Before we proceed, we note that it is customary in the bandit literature to analyse *cumulative regret*. However, the definition of cumulative regret depends on the application at hand (Kandasamy et al., 2016c) and the results in this paper can be extended to to many sensible notions of cumulative regret. However, both to simplify exposition and since our focus in this paper is optimisation, we stick to simple regret.

Assumptions: As we will be primarily focusing on continuous and compact domains and fidelity spaces, going forward we will assume, without any loss of generality, that $\mathcal{X} = [0,1]^d$ and $\mathcal{Z} = [0,1]^p$. We discuss non-continuous settings briefly at the end of Section 3. In keeping with similar work in the Bayesian optimisation literature, we will assume $g \sim \mathcal{GP}(\mathbf{0}, \kappa)$ and upon querying at (z, x) we observe $y = g(z, x) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \eta^2)$. $\kappa : (\mathcal{Z} \times \mathcal{X})^2 \to \mathbb{R}$ is the prior covariance defined on the product space. In this work, we will study exclusively κ of the following form,

$$\kappa([z, x], [z', x']) = \kappa_0 \, \phi_{\mathcal{Z}}(\|z - z'\|) \, \phi_{\mathcal{X}}(\|x - x'\|). \quad (3)$$

Here, $\kappa_0 \in \mathbb{R}_+$ is the scale parameter and $\phi_{\mathcal{Z}}, \phi_{\mathcal{X}}$ are radial kernels defined on $\mathcal{Z}, \mathcal{X}$ respectively. The fidelity space kernel $\phi_{\mathcal{Z}}$ is an important component in this work. It controls the smoothness of g across the fidelity space and hence determines how much information the lower fidelities provide about $g(z_\bullet, \cdot)$. For example, suppose that $\phi_{\mathcal{Z}}$ was a SE kernel. A favourable setting for a multi-fidelity method would be for $\phi_{\mathcal{Z}}$ to have a large bandwidth $h_{\mathcal{Z}}$ as that would imply that g is very smooth across $\mathcal{Z}$. We will see that $h_{\mathcal{Z}}$ determines the behaviour and theoretical guarantees of BOCA in a natural way when $\phi_{\mathcal{Z}}$ is the SE kernel. To formalise this notion, we will define the following function $\xi : \mathcal{Z} \to [0, 1]$.

$$\xi(z) = \sqrt{1 - \phi_{\mathcal{Z}}(\|z - z_\bullet\|)^2} \quad (4)$$

One interpretation of $\xi(z)$ is that it measures the *gap* in information about $g(z_\bullet, \cdot)$ when we query at $z \neq z_\bullet$. That is, it is the price we have to pay, in information, for querying at a cheap fidelity. Observe that ξ increases when we move away from $z_\bullet$ in the fidelity space. For the SE kernel, it can be shown[1] $\xi(z) \approx \frac{\|z - z_\bullet\|}{h_{\mathcal{Z}}}$. For large $h_{\mathcal{Z}}$, g is smoother across $\mathcal{Z}$ and we can expect the lower fidelities to be more informative about f; as expected the information gap ξ is small for large $h_{\mathcal{Z}}$. If $h_{\mathcal{Z}}$ is small and g is not smooth, the gap ξ is large and lower fidelities are not as informative.

Before we present our algorithm for the above setup, we will introduce notation for the posterior GPs for g and f. Let $\mathcal{D}_n = \{(z_i, x_i, y_i)\}_{i=1}^n$ be n fidelity, query, observation values from the GP g, where y_i was observed when evaluating $g(z_i, x_i)$. We will denote the posterior mean and standard deviation of g conditioned on $\mathcal{D}_n$ by ν_n and τ_n respectively (ν_n, τ_n can be computed from (1) by replacing $x \leftarrow [z, x]$). Therefore $g(z, x)|\mathcal{D}_n \sim \mathcal{N}(\nu_n(z, x), \tau_n^2(z, x))$ for all $(z, x) \in \mathcal{Z} \times \mathcal{X}$. We will further denote

$$\mu_n(\cdot) = \nu_n(z_\bullet, \cdot), \qquad \sigma_n(\cdot) = \tau_n(z_\bullet, \cdot), \quad (5)$$

to be the posterior mean and standard deviation of $g(z_\bullet, \cdot) = f(\cdot)$. It follows that $f|\mathcal{D}_n$ is also a GP and satisfies $f(x)|\mathcal{D}_n \sim \mathcal{N}(\mu_n(x), \sigma_n^2(x))$ for all $x \in \mathcal{X}$.

3. BOCA: Bayesian Optimisation with Continuous Approximations

BOCA is a sequential strategy to select a domain point $x_t \in \mathcal{X}$ and fidelity $z_t \in \mathcal{Z}$ at time t based on previous observations. At time t, we will first construct an upper confidence bound φ_t for the function f we wish to optimise. It takes the form,

$$\varphi_t(x) = \mu_{t-1}(x) + \beta_t^{1/2} \sigma_{t-1}(x). \quad (6)$$

Recall from (5) that μ_{t-1} and σ_{t-1} are the posterior mean and standard deviation of f using the observations from the previous $t-1$ time steps at all fidelities, i.e. the entire $\mathcal{Z} \times \mathcal{X}$ space. We will specify β_t in Theorems 1, 8. Following other UCB algorithms, our next point x_t in the domain $\mathcal{X}$ for evaluating g is a maximiser of φ_t, i.e. $x_t \in \operatorname{argmax}_{x \in \mathcal{X}} \varphi_t(x)$.

Next, we need to determine the fidelity $z_t \in \mathcal{Z}$ to query g.

[1] Strictly, $\xi(z) \leq \|z - z_\bullet\|/h_{\mathcal{Z}}$, but the inequality is tighter for larger $h_{\mathcal{Z}}$. In any case, ξ is strictly decreasing with $h_{\mathcal{Z}}$.

For this we will first select a subset $\mathcal{Z}_t(x_t)$ of $\mathcal{Z}$ as follows,

$$\mathcal{Z}_t(x_t) = \Big\{z \in \mathcal{Z} \;:\; \lambda(z) < \lambda(z_\bullet), \quad \tau_{t-1}(z, x_t) > \gamma(z), \quad \xi(z) > \beta_t^{-1/2}\|\xi\|_\infty\Big\}, \tag{7}$$

where $\gamma(z) = \sqrt{\kappa_0}\,\xi(z)\left(\frac{\lambda(z)}{\lambda(z_\bullet)}\right)^q$.

Here, ξ is the information gap function in (4) and τ_{t-1} is the posterior standard deviation of g, and p, d are the dimensionalities of $\mathcal{Z}, \mathcal{X}$. The exponent q depends on the kernel used for $\phi_{\mathcal{Z}}$. For e.g., for the SE kernel, $q = 1/(p+d+2)$. We filter out the fidelities we consider at time t using three conditions as specified above. We elaborate on these conditions in more detail in Section 3.1. If $\mathcal{Z}_t$ is not empty, we choose the cheapest fidelity in this set, i.e. $z_t \in \mathrm{argmin}_{z\in\mathcal{Z}_t} \lambda(z)$. If $\mathcal{Z}_t$ is empty, we choose $z_t = z_\bullet$.

We have summarised the resulting procedure below in Algorithm 1. An important advantage of BOCA is that it only requires specifying the GP hyper-parameters for g such as the kernel κ. In practice, this can be achieved by various effective heuristics such as maximising the GP marginal likelihood or cross validation which are standard in most BO methods. In contrast, MF-GP-UCB of Kandasamy et al. (2016a) requires tuning several other hyper-parameters.

Algorithm 1 BOCA

Input: kernel κ.

- Set $\nu_0(\cdot) \leftarrow \mathbf{0}$, $\tau_0(\cdot) \leftarrow \kappa(\cdot,\cdot)^{1/2}$, $\mathcal{D}_0 \leftarrow \varnothing$.
- for $t = 1, 2, \ldots$
  1. $x_t \leftarrow \mathrm{argmax}_{x\in\mathcal{X}}\ \varphi_t(x)$. See (6)
  2. $z_t \leftarrow \mathrm{argmin}_{z\in\,\mathcal{Z}_t(x_t)\cup\{z_\bullet\}}\ \lambda(z)$. See (7)
  3. $y_t \leftarrow$ Query g at (z_t, x_t).
  4. $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup \{(z_t, x_t, y_t)\}$. Update posterior mean ν_t, and standard deviation τ_t for g conditioned on $\mathcal{D}_t$.

3.1. Fidelity Selection Criterion

We will now provide an intuitive justification for the three conditions in the selection criterion for z_t, i.e., equation (7). The first condition, $\lambda(z) < \lambda(z_\bullet)$ is fairly obvious; since we wish to optimise $g(z_\bullet, \cdot)$ and since we are not rewarded for queries at other fidelities, there is no reason to consider fidelities that are more expensive than $z_\bullet$.

The second condition, $\tau_{t-1}(z, x_t) > \gamma(z)$ says that we will only consider fidelities where the posterior variance is larger than a threshold $\gamma(z) = \sqrt{\kappa_0}\xi(z)(\lambda(z)/\lambda(z_\bullet))^q$, which depends critically on two quantities, the cost function λ and the information gap ξ. As a first step towards parsing this condition, observe that a reasonable multi-fidelity strategy should be inclined to query cheap fidelities and learn about g before querying expensive fidelities. As $\gamma(z)$ is monotonically increasing in $\lambda(z)$, it becomes easier for a cheap z to satisfy $\tau_{t-1}(z, x_t) > \gamma(z)$ and be included in $\mathcal{Z}_t$ at time t. Moreover, since we choose z_t to be the minimiser of λ in $\mathcal{Z}_t$, a cheaper fidelity will always be chosen over expensive ones if included in $\mathcal{Z}_t$. Second, if a particular fidelity z is far away from $z_\bullet$, it probably contains less information about $g(z_\bullet, \cdot)$. Again, a reasonable multi-fidelity strategy should be discouraged from making such queries. This is precisely the role of the information gap ξ which is increasing with $\|z - z_\bullet\|$. As z moves away from $z_\bullet$, $\gamma(z)$ increases and it becomes harder to satisfy $\tau_{t-1}(z, x_t) > \gamma(z)$. Therefore, such a z is less likely to be included in $\mathcal{Z}_t(x_t)$ and be considered for evaluation. Our analysis reveals that setting γ as in (7) is a reasonable trade off between cost and information in the approximations available to us; cheaper fidelities cost less, but provide less accurate information about the function f we wish to optimise. It is worth noting that the second condition is similar in spirit to Kandasamy et al. (2016a) who proceed from a lower to higher fidelity only when the lower fidelity variance is smaller than a threshold. However, while they treat the threshold as a hyper-parameter, we are able to explicitly specify theoretically motivated values.

The third condition in (7) is $\xi(z) > \|\xi\|_\infty/\beta_t^{1/2}$. Since ξ is increasing as we move away from $z_\bullet$, it says we should exclude fidelities inside a (small) neighbourhood of $z_\bullet$. Recall that if $\mathcal{Z}_t$ is empty, BOCA will choose $z_\bullet$ by default. But when it is not empty, we want to prevent situations where we get arbitrarily close to $z_\bullet$ but not actually query *at* $z_\bullet$. Such pathologies can occur when we are dealing with a continuum of fidelities and this condition forces BOCA to pick $z_\bullet$ instead of querying very close to it. Observe that since β_t is increasing with t, this neighborhood is shrinking with time and therefore the algorithm will eventually have the opportunity to evaluate fidelities close to $z_\bullet$.

3.2. Theoretical Results

We now present our main theoretical contributions. In order to simplify the exposition and convey the gist of our results, we will only present a simplified version of our theorems. We will suppress constants, polylog terms, and other technical details that arise due to a covering argument in our proofs. A rigorous treatment is available in Appendix B.

Maximum Information Gain: Up until this point, we have not discussed much about the kernel $\phi_{\mathcal{X}}$ of the domain $\mathcal{X}$. Since we are optimising f over $\mathcal{X}$, it is natural to expect that this will appear in the bounds. Srinivas et al. (2010) showed that the statistical difficulty of GP bandits is determined by the *Maximum Information Gain* (MIG) which measures the maximum information a subset of observations have about f. We denote it by $\Psi_n(A)$ where A is a subset of $\mathcal{X}$ and n is the number of queries to f. We refer the reader to Appendix B for a formal definition of MIG. For the current exposition however, it suffices to know that for radial kernels, $\Psi_n(A)$ increases with n and the volume $\mathrm{vol}(A)$ of A. For instance, when we use an SE kernel for $\phi_{\mathcal{X}}$, we have $\Psi_n(A) \propto \mathrm{vol}(A)\, \log(n)^{d+1}$and for a Matérn kernel with smoothness parameter ν, $\Psi_n(A) \propto \mathrm{vol}(A)\, n^{1-\frac{\nu}{2\nu+d(d+1)}}$. (Srinivas et al., 2010). Let $n_\Lambda = \lfloor \Lambda/\lambda(z_\bullet)\rfloor$ denote the number of queries by a single fidelity algorithm within capital Λ. Srinivas et al. (2010) showed that the simple regret $S(\Lambda)$ for GP-UCB after capital Λ can be bounded by,

$$\text{Simple Regret for GP-UCB:} \quad S(\Lambda) \lesssim \sqrt{\frac{\Psi_{n_\Lambda}(\mathcal{X})}{n_\Lambda}}. \tag{8}$$

In our analysis of BOCA we show that most queries to g at fidelity $z_\bullet$ will be confined to a small subset of $\mathcal{X}$ which contains the optimum $x_\star$. Precisely, after capital Λ, for any $\alpha \in (0,1)$, we show there exists $\rho > 0$ such that the number of queries *outside* the following set $\mathcal{X}_\rho$ is less than n_Λ^α.

$$\mathcal{X}_\rho = \big\{x \in \mathcal{X} \,:\, f_\star - f(x) \leq 2\rho\sqrt{\kappa_0}\,\|\xi\|_\infty\big\}. \tag{9}$$

Here, ξ is from (4). While it is true that any optimisation algorithm would eventually query extensively in a neighbourhood around the optimum, a strong result of the above form is not always possible. For instance, for GP-UCB, the best achievable bound on the number of queries in any set that does not contain $x_\star$ is $n_\Lambda^{1/2}$. The fact that $\mathcal{X}_\rho$ exists relies crucially on the multi-fidelity assumptions and that our algorithm leverages information from lower fidelities when querying at $z_\bullet$. As ξ is small when g is smooth across $\mathcal{Z}$, the set $\mathcal{X}_\rho$ will be small when the approximations are highly informative about $g(z_\bullet, \cdot)$. For e.g., when $\phi_{\mathcal{Z}}$ is a SE kernel, we have $\mathcal{X}_\rho \approx \{x \in \mathcal{X} : f_\star - f(x) \leq 2\rho\sqrt{\kappa_0 p}/h_{\mathcal{Z}}\}$. When $h_{\mathcal{Z}}$ is large and g is smooth across $\mathcal{Z}$, $\mathcal{X}_\rho$ is small as the right side of the inequality is smaller. As BOCA confines most of its evaluations to this small set containing $x_\star$, we will be able to achieve much better regret than GP-UCB. When $h_{\mathcal{Z}}$ is small and g is not smooth across $\mathcal{Z}$, the set $\mathcal{X}_\rho$ becomes large and the advantage of multi-fidelity optimisation diminishes. One can similarly argue that for the Matérn kernel, as the parameter ν increases, g will be smoother across $\mathcal{Z}$, and $\mathcal{X}_\rho$ becomes smaller yielding better bounds on the regret. Below, we provide an informal statement of our main theoretical result. $\lesssim, \asymp$ will denote inequality and equality ignoring constant and polylog terms.

Theorem 1 (Informal, Regret of BOCA). *Let $g \sim \mathcal{GP}(0,\kappa)$ where κ satisfies (3). Choose $\beta_t \asymp d\log(t/\delta)$. Then, for sufficiently large Λ and for all $\alpha \in (0,1)$, there exists ρ depending on α such that the following bound holds with probability at least $1-\delta$.*

$$S(\Lambda) \;\lesssim\; \sqrt{\frac{\Psi_{n_\Lambda}(\mathcal{X}_\rho)}{n_\Lambda}} \;+\; \sqrt{\frac{\Psi_{n_\Lambda^\alpha}(\mathcal{X})}{n_\Lambda^{2-\alpha}}}$$

In the above bound, the latter term vanishes fast due to the $n_\Lambda^{-(1-\alpha/2)}$ dependence. When comparing this with (8), we see that we outperform GP-UCB by a factor of $\sqrt{\Psi_{n_\Lambda}(\mathcal{X}_\rho)/\Psi_{n_\Lambda}(\mathcal{X})} \asymp \sqrt{\mathrm{vol}(\mathcal{X}_\rho)/\mathrm{vol}(\mathcal{X})}$ asymptotically. If g is smooth across the fidelity space, $\mathcal{X}_\rho$ is small and the gains over GP-UCB are significant. If g becomes less smooth across $\mathcal{Z}$, the bound decays gracefully, but we are never worse than GP-UCB up to constant factors.

Theorem 1 also has similarities to the bounds of Kandasamy et al. (2016a) who also demonstrate better regret than GP-UCB by showing that it is dominated by queries inside a set $\mathcal{X}'$ which contains the optimum. However, their bounds depend critically on certain threshold hyper-parameters which determine the volume of $\mathcal{X}'$ among other terms in their regret. The authors of that paper note that their bounds will suffer if these hyper-parameters are not chosen appropriately, but do not provide theoretically justified methods to make this choice. In contrast, many of the design choices for BOCA fall out naturally of our modeling assumptions. Beyond this analogue, our results are not comparable to Kandasamy et al. (2016a) as the assumptions are different.

Extensions: While we have focused on continuous $\mathcal{Z}$, many of the ideas here can be extended to other settings. If $\mathcal{Z}$ is a discrete subset of $[0,1]^p$ our work extends straightforwardly. We reiterate that this will not be the same as the finite fidelity MF-GP-UCB algorithm as the assumptions are different. In particular, Kandasamy et al. (2016a) are not able to effectively share information across fidelities as we do. We also believe that Algorithm 1 can be extended to arbitrary fidelity spaces $\mathcal{Z}$ provided that a kernel can be defined on $\mathcal{Z}$. Our results can also be extended to discrete domains $\mathcal{X}$ and various other kernels for $\phi_{\mathcal{X}}$ by adopting techniques from Srinivas et al. (2010).

4. Experiments

We compare BOCA to the following four baselines: (i) GP-UCB, (ii) the GP-EI criterion in BO (Jones et al., 1998), (iii) MF-GP-UCB (Kandasamy et al., 2016a) and (iv) MF-SKO, the multi-fidelity sequential kriging optimisation method from Huang et al. (2006). All methods are based on GPs and we use the SE kernel for both the fidelity space and domain. The first two are not multi-fidelity methods, while the last two are finite multi-fidelity methods[2]. Kandasamy et al. (2016a) also study some naive multi-fidelity algorithms and demonstrate that they do not perform well; as such we will not consider such alternatives here. In all our experiments, the fidelity space was designed to be $\mathcal{Z} = [0,1]^p$ with $z_\bullet = \mathbf{1}_p = [1,\ldots,1] \in \mathbb{R}^p$ being the most expensive fi-

[2]To our knowledge, the only other work that applies to continuous approximations is Klein et al. (2015) which was developed specifically for hyper-parameter tuning. Further, their implementation is not made available and is not straightforward to implement.

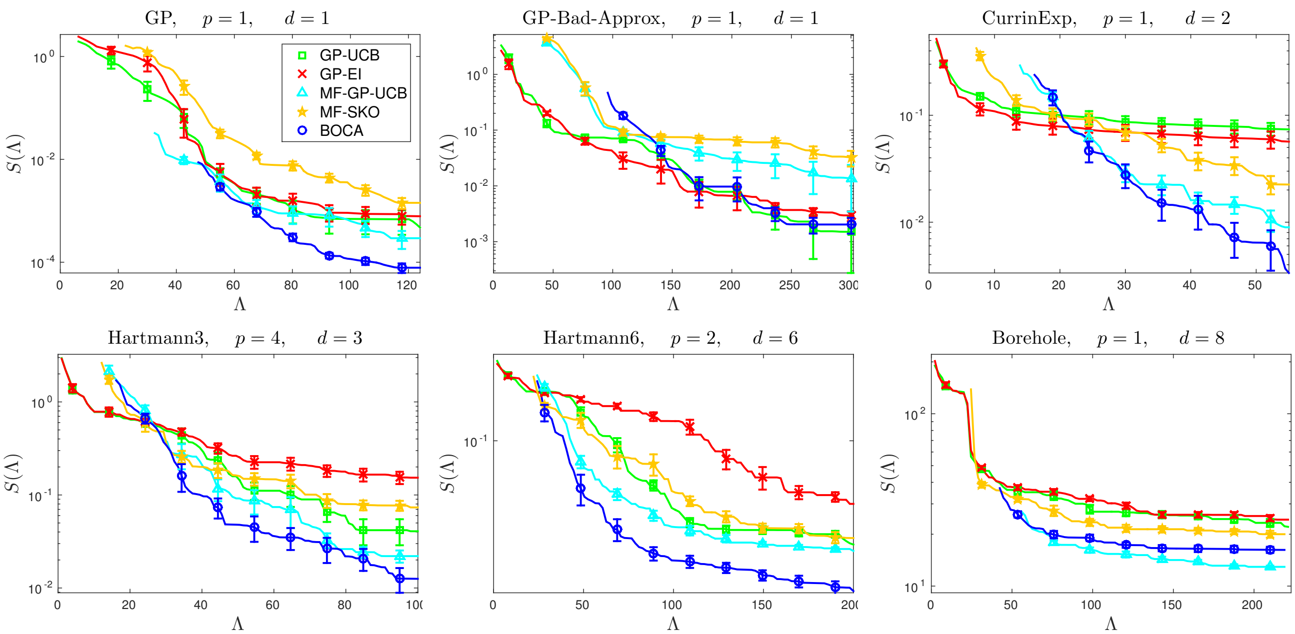


Figure 3. Results on 6 synthetic problems where we plot the simple regret $S(\Lambda)$ (lower is better) against the capital Λ. The title states the function used, and the fidelity and domain dimesions. For the first two figures we used capital $30\lambda(z_\bullet)$, therefore a method which only queries at $g(z_\bullet, \cdot)$ can make at most 30 evaluations. For the third figure we used $50\lambda(z_\bullet)$, for the fourth $100\lambda(z_\bullet)$ and for the last $200\lambda(z_\bullet)$ to reflect the dimensionality d of $\mathcal{X}$. The curves for the multi-fidelity methods start mid-way since they have not queried at $z_\bullet$ up until that point. All curves were produced by averaging over 20 experiments and the error bars indicate one standard error.

delity. For MF-GP-UCB and MF-SKO, we used 3 fidelities (2 approximations) where the approximations were obtained at $z = 0.333\mathbf{1}_p$ and $z = 0.667\mathbf{1}_p$ in $\mathcal{Z}$. Empirically, we found that both algorithms did reasonably well with 1-3 approximations, but did not perform well with a large number of approximations (> 5); even the original papers restrict experiments to 1-3 approximations. Implementation details for all methods are given in Appendix C.1.

4.1. Synthetic Experiments

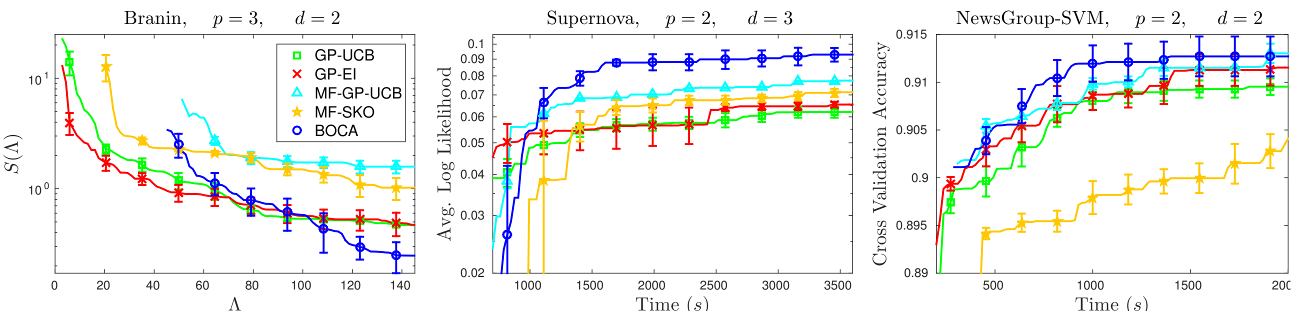
The results for the first set of synthetic experiments are given in Fig. 3. The title of each figure states the function used, and the dimensionalities p, d of the fidelity space and domain. To reflect the setting in our theory, we add Gaussian noise to the function value when observing g at any (z, x). This makes the problem more challenging than standard global optimisation problems where function evaluations are not noisy. The functions g, the cost functions λ and the noise variances η^2 are given in Appendix C.2.

The first two panels in Fig. 3 are simple sanity checks. In both cases, $\mathcal{Z} = [0, 1]$, $\mathcal{X} = [0, 1]$ and the functions were sampled from GPs. The GP was made known to all methods, i.e. all methods used the true GP in picking the next point. In the first panel, we used an SE kernel with bandwidth 0.1 for $\phi_{\mathcal{X}}$ and 1.0 for $\phi_{\mathcal{Z}}$. g is smooth across $\mathcal{Z}$ in this setting, and BOCA outperforms other baselines. The curve starts mid-way as BOCA is yet to query at $z_\bullet$ up until that point. The second panel uses the same set up as the first except we used bandwidth 0.01 for $\phi_{\mathcal{Z}}$. Even though g is highly un-smooth across $\mathcal{Z}$, BOCA does not perform poorly. This corroborates a claim that we made earlier that BOCA can naturally adapt to the smoothness of the approximations. The other multi-fidelity methods suffer in this setting.

In the remaining experiments, we use some standard benchmarks for global optimisation. We modify them to obtain g and add noise to the observations. As the kernel and other GP hyper-parameters are unknown, we learn them by maximising the marginal likelihood every 25 iterations. We outperform all methods on all problems except in the case of the Borehole function where MF-GP-UCB does better. The last synthetic experiment is the Branin function given in Fig. 4(a). We used the same set up as above, but use 10 fidelities for MF-GP-UCB and MF-SKO where the k^{th} fidelity is obtained at $z = \frac{k}{10}\mathbf{1}_p$ in the fidelity space. Notice that the performance of finite fidelity methods deteriorate. In particular, as MF-GP-UCB does not share information across fidelities, the approximations need to be designed carefully for the algorithm to work well. Our more natural modelling assumptions prevent such pitfalls. We next present two real examples in astrophysics and hyper-parameter tuning. We do *not* add noise to the observations, but treat it as optimisation tasks, where the goal is to maximise the function.

4.2. Astrophysical Maximum Likelihood Inference

We use data on TypeIa supernova for maximum likelihood inference on 3 cosmological parameters, the Hubble con-



(a) (b) (c)

Figure 4. (a): The synthetic benchmark with the Branin function where we used a capital of $50\lambda(z_\bullet)$. See caption under Fig. 3 for more details. (b), (c): Results on the supernova and news group experiments from sections 4.2 and 4.3 respectively. We have plotted the maximum value (higher is better) against wall clock time. (a) was averaged over 20 experiments while (b) and (c) were averaged over 10 xperiments each. The error bars indicate one standard error.

stant $H_0 \in (60, 80)$, the dark matter fraction $\Omega_M \in (0,1)$ and dark energy fraction $\Omega_\Lambda \in (0,1)$; hence $d = 3$. The likelihood is given by the Robertson-Walker metric, the computation of which requires a one dimensional numerical integration for each point in the dataset. Unlike typical maximum likelihood problems, here the likelihood is only accessible via point evaluations. We use the dataset from Davis et al (2007) which has data on 192 supernovae. We construct a $p = 2$ dimensional multi-fidelity problem where we can choose between data set size $N \in [50, 192]$ and perform the integration on grids of size $G \in [10^2, 10^6]$ via the trapezoidal rule. As the cost function for fidelity selection, we used $\lambda(N, G) = NG$ as the computation time is linear in both parameters. Our goal is to maximise the average log likelihood at $z_\bullet = [192, 10^6]$. For the finite fidelity methods we use three fidelities with the approximations available at $z = [97, 2.15 \times 10^3]$ and $z = [145, 4.64 \times 10^4]$ (which correspond to $0.333\mathbf{1}_p$ and $0.667\mathbf{1}_p$ after rescaling as in Section 4.1). The results are given in Fig. 4(b) where we plot the maximum average log likelihood against wall clock time as that is the cost in this experiment. The plot includes the time taken by each method to tune the GPs and determine the next points/fidelities for evaluation.

4.3. Support Vector Classification with 20 news groups

We use the 20 news groups dataset (Joachims, 1996) in a text classification task. We obtain the bag of words representation for each document, convert them to tf-idf features and feed them to a support vector classifier. The goal is to tune the regularisation penalty and the temperature of the rbf kernel both in the range $[10^{-2}, 10^3]$; hence $d = 2$. The support vector implementation was taken from scikit-learn. We set this up as a 2 dimensional multi-fidelity problem where we can choose a dataset size $N \in [5000, 15000]$ and the number of training iterations $T \in [20, 100]$. Each evaluation takes the given dataset of size N and splits it up into 5 to perform 5-fold cross validation. As the cost function for fidelity selection, we used $\lambda(N, T) = NT$ as the training/validation complexity is linear in both parameters. Our goal is to maximise the cross validation accuracy at $z_\bullet = [15000, 100]$. For the finite fidelity methods we use three fidelities with the approximations available at $z = [8333, 47]$ and $z = [11667, 73]$. The results are given in Fig. 4(c) where we plot the average cross validation accuracy against wall clock time.

5. Conclusion

We studied Bayesian optimisation with continuous approximations, by treating the approximations as arising out of a continuous fidelity space. While previous multi-fidelity literature has predominantly focused on a finite number of approximations, BOCA applies to continuous fidelity spaces and can potentially be extended to arbitrary spaces. We bound the simple regret for BOCA and demonstrate that it is better than methods such as GP-UCB which ignore the approximations and that the gains are determined by the smoothness of the fidelity space. When compared to existing multi-fidelity methods, BOCA is able to share information across fidelities effectively, has more natural modelling assumptions and has fewer hyper-parameters to tune. Empirically, we demonstrate that BOCA is competitive with other baselines in synthetic and real problems. Another nice feature of using continuous approximations is that it relieves the practitioner from having to design the approximations; she/he can specify the available approximations and let the algorithm decide how to choose them.

Going forward, we wish to extend our theoretical results to more general settings. For instance, we believe a stronger bound on the regret might be possible if $\phi_\mathcal{Z}$ is a finite dimensional kernel. Since finite dimensional kernels are typically not radial (Sriperumbudur et al., 2016), our analysis techniques will not carry over straightforwardly. Another line of work that we have alluded to is to study more general fidelity spaces with an appropriately defined kernel $\phi_\mathcal{Z}$.

Acknowledgements

We would like to thank Renato Negrinho for reviewing an initial draft of this paper. This research is supported in part by DOE grant DESC0011114 and NSF grant IIS1563887. KK is supported by a Facebook Ph.D. fellowship.

References

Agarwal, Alekh, Duchi, John C, Bartlett, Peter L, and Levrard, Clement. Oracle inequalities for computationally budgeted model selection. In *COLT*, 2011.

Auer, Peter. Using Confidence Bounds for Exploitation-exploration Trade-offs. *J. Mach. Learn. Res.*, 2003.

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 RL. *CoRR*, 2010.

Cutler, Mark, Walsh, Thomas J., and How, Jonathan P. Reinforcement Learning with Multi-Fidelity Simulators. In *ICRA*, 2014.

Davis et al, T. M. Scrutinizing Exotic Cosmological Models Using ESSENCE Supernova Data Combined with Other Cosmological Probes. *Astrophysical Journal*, 2007.

de Freitas, Nando, Smola, Alex J., and Zoghi, Masrour. Exponential Regret Bounds for Gaussian Process Bandits with Deterministic Observations. In *ICML*, 2012.

Forrester, Alexander I. J., Sóbester, András, and Keane, Andy J. Multi-fidelity optimization via surrogate modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 2007.

Ghosal, Subhashis and Roy, Anindya. Posterior consistency of Gaussian process prior for nonparametric binary regression". *Annals of Statistics*, 2006.

Gonzalez, J., Longworth, J., James, D., and Lawrence, N. Bayesian Optimization for Synthetic Gene Design. In *BayesOpt*, 2014.

Hernández-Lobato, José Miguel, Hoffman, Matthew W, and Ghahramani, Zoubin. Predictive Entropy Search for Efficient Global Optimization of Black-box Functions. In *NIPS*, 2014.

Huang, D., Allen, T.T., Notz, W.I., and Miller, R.A. Sequential kriging optimization using multiple-fidelity evaluations. *Structural and Multidisciplinary Optimization*, 2006.

Hutter, Frank, Hoos, Holger H., and Leyton-Brown, Kevin. Sequential Model-based Optimization for General Algorithm Configuration. In *LION*, 2011.

Joachims, Thorsten. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. Technical report, DTIC Document, 1996.

Jones, D. R., Perttunen, C. D., and Stuckman, B. E. Lipschitzian Optimization Without the Lipschitz Constant. *J. Optim. Theory Appl.*, 1993.

Jones, Donald R., Schonlau, Matthias, and Welch, William J. Efficient global optimization of expensive black-box functions. *J. of Global Optimization*, 1998.

Kandasamy, Kirthevasan, Schenider, Jeff, and Póczos, Barnabás. High Dimensional Bayesian Optimisation and Bandits via Additive Models. In *International Conference on Machine Learning*, 2015.

Kandasamy, Kirthevasan, Dasarathy, Gautam, Oliva, Junier, Schenider, Jeff, and Póczos, Barnabás. Gaussian Process Bandit Optimisation with Multi-fidelity Evaluations. In *Advances in Neural Information Processing Systems*, 2016a.

Kandasamy, Kirthevasan, Dasarathy, Gautam, Oliva, Junier B, Schneider, Jeff, and Poczos, Barnabas. Multi-fidelity gaussian process bandit optimisation. *arXiv preprint arXiv:1603.06288*, 2016b.

Kandasamy, Kirthevasan, Dasarathy, Gautam, Schneider, Jeff, and Poczos, Barnabas. The Multi-fidelity Multi-armed Bandit. In *NIPS*, 2016c.

Klein, A., Bartels, S., Falkner, S., Hennig, P., and Hutter, F. Towards efficient Bayesian Optimization for Big Data. In *BayesOpt*, 2015.

Lam, Rémi, Allaire, Douglas L, and Willcox, Karen E. Multifidelity optimization using statistical surrogate modeling for non-hierarchical information sources. In *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, pp. 0143, 2015.

Li, Lisha, Jamieson, Kevin, DeSalvo, Giulia, Rostamizadeh, Afshin, and Talwalkar, Ameet. Hyperband: A novel bandit-based approach to hyperparameter optimization. *arXiv preprint arXiv:1603.06560*, 2016.

Lizotte, Daniel, Wang, Tao, Bowling, Michael, and Schuurmans, Dale. Automatic gait optimization with gaussian process regression. In *IJCAI*, 2007.

Martinez-Cantin, R., de Freitas, N., Doucet, A., and Castellanos, J. Active Policy Learning for Robot Planning and Exploration under Uncertainty. In *Proceedings of Robotics: Science and Systems*, 2007.

Mockus, Jonas. Application of Bayesian approach to numerical methods of global and stochastic optimization. *Journal of Global Optimization*, 1994.

Parkinson, D., Mukherjee, P., and Liddle, A.. R. A Bayesian model selection analysis of WMAP3. *Physical Review*, 2006.

Poloczek, Matthias, Wang, Jialei, and Frazier, Peter I. Multi-information source optimization. *arXiv preprint arXiv:1603.00389*, 2016.

Rasmussen, C.E. and Williams, C.K.I. *Gaussian Processes for Machine Learning*. UPG Ltd, 2006.

Sabharwal, A, Samulowitz, H, and Tesauro, G. Selecting near-optimal learners via incremental data allocation. In *AAAI*, 2015.

Seeger, MW., Kakade, SM., and Foster, DP. Information Consistency of Nonparametric Gaussian Process Methods. *IEEE Transactions on Information Theory*, 2008.

Snoek, J., Larochelle, H., and Adams, R. P. Practical Bayesian Optimization of Machine Learning Algorithms. In *NIPS*, 2012.

Srinivas, Niranjan, Krause, Andreas, Kakade, Sham, and Seeger, Matthias. Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design. In *ICML*, 2010.

Sriperumbudur, Bharath et al. On the optimal estimation of probability measures in weak and strong topologies. *Bernoulli*, 22(3):1839–1893, 2016.

Swersky, Kevin, Snoek, Jasper, and Adams, Ryan P. Multi-task bayesian optimization. In *NIPS*, 2013.

Thompson, W. R. On the Likelihood that one Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Biometrika*, 1933.

Xiong, Shifeng, Qian, Peter Z. G., and Wu, C. F. Jeff. Sequential design and analysis of high-accuracy and low-accuracy computer codes. *Technometrics*, 2013.

Zhang, C. and Chaudhuri, K. Active Learning from Weak and Strong Labelers. In *NIPS*, 2015.