Title

Bayesian Optimization Via Barrier Functions

Journal

Journal of Computational and Graphical Statistics, 31(1)

ISSN

1061-8600

Authors

Pourmohamad, Tony

Lee, Herbert KH

Publication Date

2022-01-02

DOI

10.1080/10618600.2021.1935270

Bayesian Optimization via Barrier Functions

Tony Pourmohamad
Genentech, Inc.
and
Herbert K. H. Lee
Department of Statistics,
University of California, Santa Cruz

Abstract

Hybrid optimization methods that combine statistical modeling with mathematical programming have become a popular solution for Bayesian optimization because they can better leverage both the efficient local search properties of the numerical method and the global search properties of the statistical model. These methods seek to create a sequential design strategy for efficiently optimizing expensive black-box functions when gradient information is not readily available. In this article, we propose a novel Bayesian optimization strategy that combines response surface modeling with barrier methods to efficiently solve expensive constrained optimization problems in computer modeling. At the heart of all Bayesian optimization algorithms is an acquisition function for effectively guiding the search. Our hybrid algorithm is guided by a novel acquisition function that tries to decrease the objective function as much as possible while simultaneously trying to ensure that the boundary of the constraint space is never crossed. Illustrations highlighting the success of our method are provided, including a real-world computer model optimization experiment from hydrology. Supplementary materials for this article are available online.

Keywords: Black-box function, expensive computer experiments, Gaussian process

1 Introduction

Constrained optimization problems are pervasive in scientific and industrial endeavors. In many engineering applications, physical systems of interest are often represented as black-box functions, and these black-box functions can be difficult to optimize because their outputs may be complex, multi-modal, and difficult to understand. The problem becomes even more challenging when the black-box functions are computationally expensive to evaluate and no gradient information is available, as well as when the constraint boundaries are not known in advance and are nonlinear. Bayesian optimization (BO) has emerged as a powerful tool for solving global optimization problems of expensive black-box functions (Jones et al., 1998). Having origins in the work of Mockus et al. (1978), BO is an efficient sequential design strategy for optimizing black-box functions, in as few steps as possible, that does not require gradient information (Brochu et al., 2010). The success of BO has been heavily tied to the use of acquisition functions for guiding the search (Taddy et al., 2009; Snoek et al., 2012; Lindberg and Lee, 2015; Eriksson et al., 2019). An appropriate acquisition function should accurately encode the beliefs about which is the best next input to evaluate, while also striking a balance between exploration (global search) and exploitation (local search). It is due to these reasons that we develop a novel BO acquisition function that is capable of reliably guiding the search algorithm, with few function evaluations, to the global solution of a black-box constrained optimization problem. When the black-box function is expensive to evaluate, it is particularly important to be able to use as few function evaluations as possible.

In this article, we seek to solve problems of the form

$$\min_{x}\{f(x) : c(x) \leq 0,\, x \in \mathcal{X}\} \tag{1}$$

where $\mathcal{X} \subset \mathbb{R}^d$ is a known, bounded region such that $f : \mathcal{X} \to \mathbb{R}$ denotes a scalar-valued objective function and $c : \mathcal{X} \to \mathbb{R}^m$ denotes a vector of m constraint functions. Both the objective, f, and constraint functions, c, are assumed to be expensive black-box functions, and we focus on the derivative-free situation where no information about the gradients of the objective and constraint functions is available (Conn et al., 2009). We also define the feasible set of points, $\mathcal{F} \subset \mathcal{X}$, to be the collection of inputs x that satisfy the constraint

functions c. Lastly, we make the assumption that a solution to (1) exists.

Provably convergent methods for solving derivative-free constrained optimization problems are plentiful in the mathematical programming literature (Conn et al., 2009), yet their search is typically focused locally and so only local solutions can be guaranteed. On the other hand, statistical models offer the opportunity to search the space globally for solutions to constrained optimization problems, but suffer from a lack of convergence guarantees, speed as compared to local search algorithms, and typically heuristics are needed to handle constraints. However, although not coined as BO, many authors have realized that the marriage of mathematical programming with statistical modeling could serve to better leverage both the efficient local search properties of the numerical method and the global search properties of the statistical model. For example, Gramacy et al. (2016) took a hybrid optimization approach and combined statistical surrogate modeling with a penalty function approach to derive an acquisition function based on augmented Lagrangians. Likewise, Pourmohamad and Lee (2020) combined statistical surrogate modeling with a filter method in order to derive an acquisition function that chose inputs that maximized the probability that a point would be acceptable to the filter and thus reduce the objective function. In this article, we take a similar position and derive a novel acquisition function based on the hybridization of Gaussian process surrogate modeling (Santner et al., 2003) and barrier methods (Nocedal and Wright, 2006), that tries to decrease the objective function as much as possible while also simultaneously trying to ensure that the constraint is satisfied. Our new BO approach is competitive with the state-of-the-art current methods.

The remainder of this article is organized as follows. In Section 2, we introduce the three major components that we hybridize for our BO algorithm. Section 3 explains the derivation of our novel acquisition function. Two versions of the acquisition function are proposed, and we highlight the rationale behind each. Section 4 demonstrates the efficiency of our new BO algorithm by solving two synthetic test problems and a real-world hydrology computer experiment. Lastly, Section 5 finishes with some discussion.

2 Hybrid Optimization

Section 2 introduces the three components of our algorithm that we hybridize in order to solve problems of the form (1).

2.1 Gaussian Process Surrogate Modeling

Popular in the modeling of computer experiments, surrogate models are efficient statistical models that serve as a fast approximation to the true computer model or black-box function (Santner et al., 2003; Kleijnen, 2015; Gramacy, 2020). Due to their analytical tractability, the canonical choice for modeling of computer experiments has been the Gaussian process (GP). GPs are distributions over functions such that the joint distribution at any finite set of points is a multivariate Gaussian distribution, and are defined by a mean function and a covariance function. GPs have a number of desirable properties such as being flexible (a form of nonparametric regression), being able to closely approximate most functions, and often being much cheaper/faster to evaluate than the actual computer model. More importantly, using GPs for surrogate modeling allows for uncertainty quantification of computer models (or black-box functions) at untried (or unobserved) inputs. Let $\{x^{(i)}, y^{(i)}\}_{i=1}^n$ denote the input-output pairs of data after n evaluations of a computer model. The GP, $Y(x)$, serves as a flexible regression model for the data $\{x^{(i)}, y^{(i)}\}_{i=1}^n$ and its predictive equations arise as a simple application of conditioning for multivariate normal joint distributions, i.e., the predictive distribution $Y(x)|\{x^{(i)}, y^{(i)}\}_{i=1}^n$ at a new input x follows another Gaussian process $Y(x)|\{x^{(i)}, y^{(i)}\}_{i=1}^n \sim N(\mu(x), \sigma^2(x))$.

2.2 Barrier Methods

Barrier methods (Nocedal and Wright, 2006), also known as interior point methods, are a natural strategy for solving problems of the form (1) as they try to decrease the objective function as much as possible while ensuring that the boundary of the feasible set $\mathcal{F}$ is never crossed. In order to ensure that the boundary is never crossed, barrier methods replace the inequality constraints with an extra term in the objective function that can be viewed as

a penalty for approaching the boundary. And so, we can rewrite (1) as

$$\min_x \left\{ f(x) + \sum_{i=1}^{m} \mathbf{B}_{\{c_i(x)\leq 0\}}(x) \right\} \tag{2}$$

where $\mathbf{B}_{\{c_i(x)\leq 0\}}(x) = 0$ if $c_i(x) \leq 0$ and ∞ otherwise. In general, this reformulation is not particularly useful as it introduces an abrupt discontinuity when $c_i(x) > 0$. However, we can replace the discontinuous function in (2) with a continuous approximation, $\phi(x)$, that is ∞ when $c_i(x) > 0$ but is finite for $c_i(x) \leq 0$ and approaches ∞ as $c_i(x)$ approaches zero. The continuous approximation $\phi(x)$, known as the barrier function, thereby creates a "barrier" to exiting the feasible region. A typical choice of barrier function is the log barrier function which is defined as

$$\phi(x) = -\left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \log(-c_i(x)) \tag{3}$$

for $\gamma > 0$. Using the log barrier function, we can approximate the problem in (2) as

$$\min_x \{B(x;\gamma)\} = \min_x \left\{ f(x) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \log(-c_i(x)) \right\}. \tag{4}$$

Here we note that for $c_i(x) < 0$, $\phi(x)$ is a smooth approximation of $\sum_{i=1}^{m} \mathbf{B}_{\{c_i(x)\leq 0\}}(x)$, and that this approximation improves as γ goes to ∞.

2.3 Bayesian Optimization

A method that dates back to Mockus et al. (1978), Bayesian optimization (BO) is a sequential design strategy for efficiently optimizing black-box functions, in few steps, that does not require gradient information (Brochu et al., 2010). More specifically, BO seeks to solve the minimization problem

$$x^* \in \underset{x\in\mathcal{X}}{\operatorname{argmin}}\, f(x). \tag{5}$$

The minimization problem in (5) is solved by iteratively developing a statistical surrogate model of the unknown objective function f, and at each step of this iterative process, using predictions from the statistical surrogate model to maximize an acquisition (or utility) function, $a(x)$, that measures how promising each location in the input space, $x \in \mathcal{X}$, is if

it were to be the next chosen point to evaluate. Thus, the role of the acquisition function, $a(x)$, is to guide the search for the solution to (5). We introduce new acquisition functions in Section 3, and review some existing examples here, as different choices of acquisition functions should lead to different measures of belief of the search algorithm when searching for the best next input to evaluate. Bayesian optimization essentially embeds a cheaper optimization problem inside of a difficult and expensive outer optimization problem, and so a good acquisition function should be easy to evaluate and quick to maximize with respect to the original outer optimization problem. GPs have been the typical choice of statistical surrogate model for the objective function f in BO, and this is due to their flexibility, well-calibrated uncertainty, and analytic properties (Gramacy, 2020).

Lastly, although the general definition of BO is that of an unconstrained optimization problem, extensions to the constrained optimization case are straightforward and many (Gardner et al., 2014; Gramacy et al., 2016; Letham et al., 2019).

2.3.1 Expected Improvement

Originally introduced in the computer modeling literature (Jones et al., 1998), the expected improvement (EI) acquisition function has become one of the most famous, and probably most used, acquisition functions in BO. Realizing the importance of the exploration-exploitation tradeoff, Jones et al. (1998) defined the improvement statistic at a proposed input x to be $I(x) = \max_x\{0, f^n_{\min} - Y(x)\}$ where, after n runs of the computer model, $f^n_{\min} = \min\{f(x_1), ..., f(x_n)\}$ is the current minimum value observed amongst all feasible points. Since the proposed input x has not yet been observed, $Y(x)$ is unknown and can be regarded as a random variable. Likewise, $I(x)$ can be regarded as a random variable and so new candidate inputs, x^*, can be selected by maximizing the expected improvement, i.e.,

$$x^* \in \arg\max_{x\in\mathcal{X}} \mathbb{E}\{I(x)\}. \tag{6}$$

If we treat $Y(x)$ as coming from a GP then, conditional on a particular parameterization of the GP, the expected improvement acquisition function is available in closed form as

$$\mathbb{E}(I(x)) = (f^n_{\min} - \mu^n(x))\Phi\left(\frac{f^n_{\min} - \mu^n(x)}{\sigma^n(x)}\right) + \sigma^n(x)\phi\left(\frac{f^n_{\min} - \mu^n(x)}{\sigma^n(x)}\right) \tag{7}$$

where $\mu^n(x)$ and $\sigma^n(x)$ are the mean and standard deviation of the predictive distribution of $Y(x)$, and $\Phi(\cdot)$ and $\phi(\cdot)$ are the standard normal cdf and pdf respectively. The equation in (7) provides a combined measure of how promising a candidate point is, that trades off between local search ($\mu(x)$ under $f_{\min}$) and global search ($\sigma(x)$).

Extending EI to the constrained optimization case, Schonlau et al. (1998) defined the constrained expected improvement (CEI) as

$$\text{CEI}(x) = E\{I(x)\} \times \Pr(c(x) \leq 0) \tag{8}$$

where $\Pr(c(x) \leq 0)$ is the probability of satisfying the joint constraints. Here, $I(x)$ uses an $f^n_{\min}$ defined over the region where the constraint functions are satisfied. Again, new candidate inputs, x^*, can now be selected by maximizing the expected feasible improvement, i.e.,

$$x^* \in \arg\max_{x \in \mathcal{X}} \mathbb{E}\{I(x)\} \times \Pr(c(x) \leq 0). \tag{9}$$

Here the formula in (7) still holds, however, we are now weighting EI by the probability that x is feasible.

2.3.2 Augmented Lagrangian

Gramacy et al. (2016) introduced augmented Lagrangian (AL) methods as a means of solving constrained BO problems. Similar to barrier methods, the AL method (see, e.g., Nocedal and Wright (2006)) takes the constrained optimization problem in (1) and turns it into an unconstrained optimization problem by means of a penalty parameter, i.e.,

$$L_A(x; \lambda, \rho) = f(x) + \lambda^T c(x) + \frac{1}{2\rho} \sum_{j=1}^{m} \max(0, c_j(x))^2. \tag{10}$$

Here, $\rho > 0$ is a penalty parameter and $\lambda \in \mathbb{R}^m_+$ serves the role of the Lagrange multiplier. Solving the new unconstrained problem in (10) proceeds iteratively where, given the current values of ρ^{k-1} and λ^{k-1}, at iteration k one approximately solves

$$x^* \in \operatorname*{argmin}_{x \in \mathcal{X}} L_A(x; \lambda^{k-1}, \rho^{k-1}). \tag{11}$$

One of the ways that Gramacy et al. (2016) extended AL methods to the BO framework was by modeling the augmented Lagrangian in (10) using independent GP surrogates $Y_f(x)$ and $Y_c(x) = (Y_{c_1}(x), ..., Y_{c_m}(x))$ for the objective and constraint functions, i.e.,

$$Y(x) = Y_f(x) + \lambda^T Y_c(x) + \frac{1}{2\rho} \sum_{j=1}^{m} \max(0, Y_{c_j}(x))^2. \tag{12}$$

Solving for x^* can now proceed, say, by selecting the point that minimizes the predictive mean surface $E(Y(x))$. For other strategies within the AL framework for solving for x^* via BO, please refer to Gramacy et al. (2016) and Picheny et al. (2016).

3 Novel Acquisition Functions

As discussed previously, the heart of all Bayesian optimization algorithms is an acquisition function, $a(x)$, for effectively guiding the search. It is also important that the acquisition function should balance exploration — improving the model in the less explored parts of the search space, and exploitation — favoring parts the model predicts as promising. In what follows, we explain the derivation of our novel acquisition function, a hybridization of the methods in Section 2, and explore two different variations of the acquisition function.

3.1 Expected Barrier Method

One of the simplest approaches to hybridizing mathematical programming with statistical modeling is to build a surrogate model based on the outputs of the mathematical program, i.e., modeling $y^{(i)} = B(x^{(i)}; \gamma)$ via $f^{(i)}$ and $c^{(i)}$ by fitting a GP surrogate model, $Y(x)$, to the n pairs $\{x^{(i)}, y^{(i)}\}_{i=1}^n$. However, as pointed out in Gramacy et al. (2016), models such as these will likely require nonstationary GP surrogate models in order to do a good job at model fitting and prediction which ultimately will affect how well we are able to maximize our acquisition function since this function will critically rely on the GP surrogate predictions. Instead, we follow the recommendation of Gramacy et al. (2016) and model the components of the barrier method, i.e. f and c, separately using independent surrogate models. We note that the use of correlated surrogate models for f and c may yield improvements (Pourmohamad and Lee, 2016), although we found that using inde-

pendent GP surrogate models worked about as well in practice on this problem and were faster and easier to implement. Working with independent surrogate models $Y_f(x)$ and $Y_c(x) = (Y_{c_1}(x), ..., Y_{c_m}(x))$ for the objective and constraint functions, respectively, we can model $y^{(i)} = B(x^{(i)}; \gamma)$ with the following surrogate model

$$Y(x) = Y_f(x) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \log(-Y_{c_i}(x)). \tag{13}$$

Optimization can now proceed by searching the predictive mean surface of $Y(x)$. In order to do so, we look to minimize the expectation of $Y(x)$, i.e.,

$$\begin{aligned}
\min_x \mathbb{E}(Y(x)) &= \min_x \mathbb{E}\left(Y_f(x) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \log(-Y_{c_i}(x))\right) \\
&= \min_x \mathbb{E}\left(Y_f(x)\right) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \mathbb{E}\left(\log(-Y_{c_i}(x))\right) \\
&\approx \min_x \mathbb{E}\left(Y_f(x)\right) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \left(\log(\mathbb{E}(-Y_{c_i}(x))) - \frac{\mathbb{V}(-Y_{c_i}(x))}{2\mathbb{E}(-Y_{c_i}(x))^2}\right) \\
&= \min_x \mu_f - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \left(\log(-\mu_{c_i}) + \frac{\sigma^2_{c_i}}{2\mu^2_{c_i}}\right)
\end{aligned} \tag{14}$$

The derivation of the expectation of the log operator, in the third line of (14), is taken from Teh et al. (2007) and is a direct consequence of taking a second order Taylor expansion about $\mathbb{E}(-Y_{c_i(x)})$ (see Appendix A for derivation). Now, it is clear to see that minimizing the predictive mean in (14) can be viewed as maximizing the following acquisition function:

$$a(x) = -\mu_f + \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \left(\log(-\mu_{c_i}) + \frac{\sigma^2_{c_i}}{2\mu^2_{c_i}}\right). \tag{15}$$

Following the logic in Section 2.3, we can now sequentially optimize this novel acquisition function in order to guide our search for the solution to (1). As straight forward as this may seem, careful inspection of the acquisition function in (15) reveals two non-trivial challenges that must be addressed before its use. The first challenge is that γ is a free parameter that, in the context of Bayesian optimization, has no explicit rules in how it must be set. As we will highlight in the subsequent section, care must be taken when choosing the value of γ. The second challenge, or rather undesirable characteristic, of our

acquisition function is that there is no variability associated with the objective function in it, but only with the constraints, i.e., $\sigma^2_{c_i}$. Without a term like σ^2_f in (15) to measure our prediction uncertainty for the objective function, our acquisition function will tend to favor exploitation, rather than exploration, as it will assume that we are predicting the objective function at untried inputs exactly correctly. In what follows for the remainder of Section 3, we explore solutions to these two challenges and further validate these solutions in Section 4.

3.2 The Role of γ

In the mathematical programming literature (e.g., Nocedal and Wright (2006)), it is common practice to have the value of $\gamma \to \infty$ such that, at iteration $k+1$ of the barrier method, $\gamma_{k+1} > \gamma_k$. In effect, this leads to steadily decreasing the penalty for approaching the boundary of the feasible set throughout the optimization. Conceptually, this means that the optimization algorithm starts with a large buffer on the edge of the feasible set, that in effect smooths out the feasible set. Once the algorithm settles to a fixed point on that buffer, the penalty is decreased which in effect decreases the buffer and thus allows for the algorithm to penetrate deeper towards the edge of the feasible set.

Nocedal and Wright (2006) give heuristics for the choice of schedule for γ where starting with a large penalty and then decreasing it helps with handling the nonlinearity in the penalized function and avoiding getting stuck in a local optimum of the constraint. In a similar fashion, we recommend an approach that reduces the number of arbitrary decisions the user needs to make, and so we allow for the current evaluated data, $\{x^{(i)}, f^{(i)}, c^{(i)}\}_{i=1}^n$, to be used to choose the appropriate value of γ dynamically. To this end, we propose allowing γ to be defined as $\gamma = 1/\sigma^2_f$, where σ^2_f is the predictive variance associated with the surrogate model for the objective function f. Setting γ this way reflects the fact that we think that the exploration of the objective function's surface should be based on our level of certainty about it. Here, σ^2_f will be large in areas of the space that do not have many data points, and small in areas of the space that already have many data points. Thus, in a local region of the space, as we accumulate more data points in that area, the barrier penalty, $1/\gamma$, will naturally be decreased, and we will be able to push closer to the

boundary, just as recommend by Nocedal and Wright (2006).

Under the choice of $\gamma = 1/\sigma_f^2$, we obtain the updated acquisition function

$$a(x) = -\mu_f + \sigma_f^2 \sum_{i=1}^{m} \left(\log(-\mu_{c_i}) + \frac{\sigma_{c_i}^2}{2\mu_{c_i}^2} \right). \tag{16}$$

We refer to this acquisition function as "One Over Sigma Squared" (OOSS). The OOSS acquisition function incorporates the uncertainty in both the objective and constraint functions, and is able to adaptively set the penalty according to the local number of data points.

3.3 Expected Improvement Approach

Although the EI acquisition function was originally developed for the case of unconstrained optimization, we can exploit its natural exploration-exploitation characteristics by inserting the improvement function into the minimization problem in (14). Moreover, the use of the improvement function will also allow us to naturally incorporate our uncertainty in the prediction of the objective function into our acquisition function via the uncertainty term σ_f. Replacing the objective function's surrogate model, $Y_f(x)$, with the improvement function $-I(x)$ in (13), yields

$$\min_x \mathbb{E}\left(-I(x) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \log(-c_i(x)) \right) = \min_x -\mathbb{E}(I(x)) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \left(\log(-\mu_{c_i}) + \frac{\sigma_{c_i}^2}{2\mu_{c_i}^2} \right). \tag{17}$$

Note that since we are minimizing in (14) we will need to use the negative improvement function. The minimization problem in (17) leads to the following acquisition function

$$a(x) = (f_{\min}^n - \mu_f)\Phi\left(\frac{f_{\min}^n - \mu_f}{\sigma_f} \right) + \sigma_f \phi\left(\frac{f_{\min}^n - \mu_f}{\sigma_f} \right) + \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \left(\log(-\mu_{c_i}) + \frac{\sigma_{c_i}^2}{2\mu_{c_i}^2} \right). \tag{18}$$

Similar to the OOSS acquisition function, we allow for γ to be chosen adaptively by setting $\gamma = 1/\sigma_f^2$ in (18). For the remainder of the article we refer to this acquisition function as the EI-OOSS acquisition function.

4 Illustrative Examples

More and more test problems and comparators have become available in the literature as Bayesian optimization becomes a more relevant tool for solving constrained optimization problems. To demonstrate the effectiveness of our novel acquisition function, we solve two constrained optimization problems from the literature (Gramacy et al., 2016; Pourmohamad and Lee, 2020), as well as a constrained optimization problem with no Bayesian optimization comparators. Two of the three problems are synthetic problems where the exact solutions to the problems are known, and the third problem is motivated by a real-world hydrology computer experiment that requires running an expensive black-box computer model. We solved each of the three problems using the two variations of the proposed acquisition function (Section 3.2 and 3.3) in order to compare and contrast them, and also included both the augmented Lagrangian (AL) approach (Gramacy et al., 2016) and the constrained expected improvement (CEI) approach (Schonlau et al., 1998) as comparators. Where available, we included additional comparator results from the literature. Lastly, there are many tools and software packages available for fitting GPs to data; for all of our examples we used the `R` package `laGP` (Gramacy, 2016) when fitting our GP surrogate models to the objective and constraint functions.

4.1 Modified Townsend Problem

The modified Townsend problem (Townsend, 2014) is a constrained optimization problem that is not new to the mathematical community, but to the best of our knowledge has not been solved from a Bayesian optimization point-of-view. The modified Townsend problem is defined as follows:

$$\begin{aligned} \min \ & f(x_1, x_2) = -(\cos((x_1 - 0.1)x_2))^2 - x_1 \sin(3x_1 + x_2) \\ \text{s.t. } & c(x_1, x_2) = x_1^2 + x_2^2 - \left(2\cos(t) - \frac{1}{2}\cos(2t) - \frac{1}{4}\cos(3t) - \frac{1}{8}\cos(4t)\right)^2 - (2\sin(t))^2 \end{aligned} \tag{19}$$

where $t = \arctan(x_1/x_2)$, $-2.25 \leq x_1 \leq 2.5$, and $-2.5 \leq x_2 \leq 1.75$. The optimal solution to the modified Townsend problem is $f(x_1, x_2) = -2.0239884$, which occurs at $(x_1, x_2) = (2.0052938, 1.1944509)$. The modified Townsend problem is a low dimensional problem

having only two inputs, x_1 and x_2, however, solving the problem is nontrivial as both the objective and constraint functions are highly nonlinear, and the solution to the problem is known to lie along the boundary of the feasible set $\mathcal{F}$ (Figure 1). The problem is further complicated as there are several local minima within the feasible set which can trap local or greedy search algorithms.

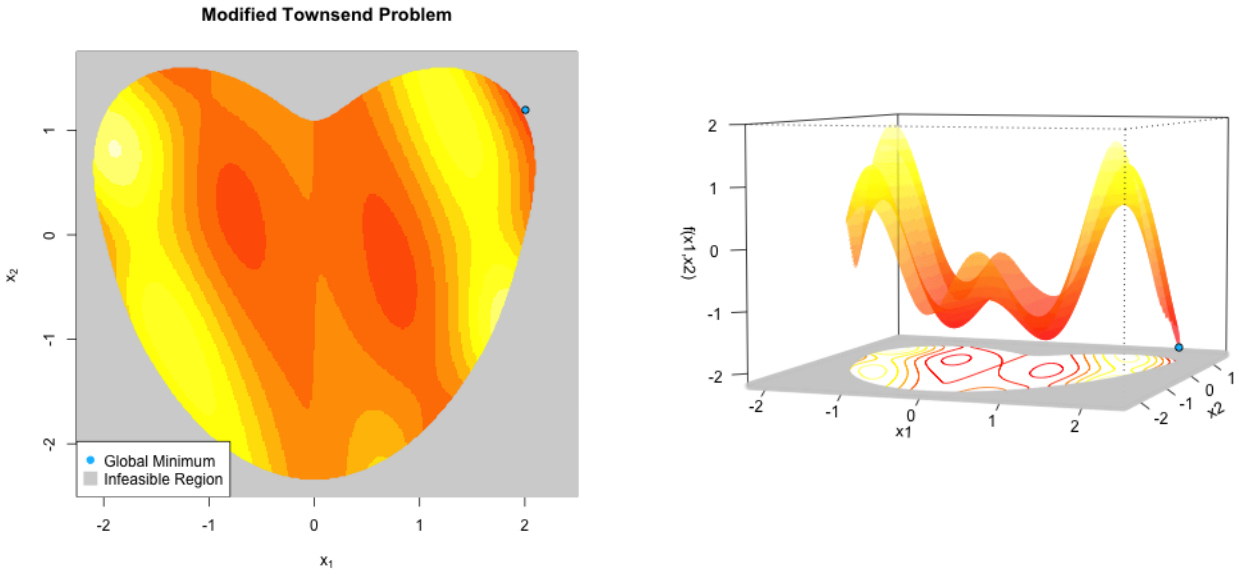
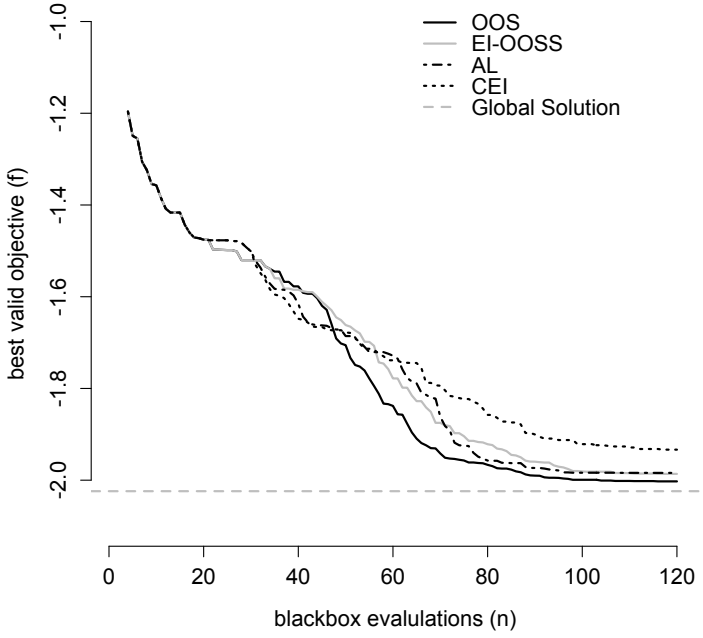


Figure 1: A view of the the objective function of the modified Townsend problem subject to the constraint function. The problem contains several local minima, and the global minimum is known to lie along the boundary of the feasible space.

To solve the modified Townsend problem, we start with an initial random sample of 20 inputs from a Latin hypercube design (LHD) (McKay et al., 1979) over the input space and sequentially choose 100 more inputs by following the BO paradigm and using the two variations of acquisitions functions found in Section 3 (i.e., OOSS and EI-OOSS), as well as the AL and CEI approaches for comparison. For each of the acquisition functions, we conduct 30 repetitions of a Monte Carlo experiment in order to understand the distribution and robustness of our solutions for the modified Townsend problem. Note that each Monte Carlo experiment is initialized using a LHD of size 20. These initial designs are kept the same across each acquisition function.



n	45	70	120
	95%		
OOSS	-1.352	-1.831	-1.963
EI-OOSS	-1.352	-1.670	-1.933
AL	-1.372	-1.592	-1.930
CEI	-1.563	-1.635	-1.747
	average		
OOSS	-1.620	-1.942	-2.003
EI-OOSS	-1.611	-1.875	-1.986
AL	-1.638	-1.733	-1.983
CEI	-1.664	-1.765	- 1.920
	5%		
OOSS	-1.942	-2.018	-2.021
EI-OOSS	-1.942	-2.002	-2.014
AL	-1.961	-1.964	-2.011
CEI	-1.897	-2.002	-2.002

Figure 2: The results of running 30 Monte Carlo repetitions with random starting inputs. The plot and table show the average best valid objective function values found over 120 black-box iterations. 5^{th} and 95^{th} percentiles are also included to better understand the spread of the distribution on the Monte Carlo repetitions. Here the horizontal dashed line corresponds to the global solution (i.e., minimum) of the optimization problem.

On average, both the OOSS and EI-OOSS acquisition functions were able to find the global solution of the problem over the additional 100 updates (Figure 2), with OOSS being much better overall at decreasing the objective function than EI-OOSS, AL and CEI. AL and CEI seem to do a better job at minimizing the problem at the early stages as compared to OOSS and EI-OOSS, however, the OOSS acquisition function does steadily decrease the objective function in the search for the global minimum and, at around 50 iterations, has caught up to all of the BO algorithms and in the end has consistently found the global solution of the problem. Likewise, EI-OOSS and AL seem to converge to the global solution

of the problem on average as well but at a slightly slower rate than the OOSS acquisition function. The same cannot be said for the CEI as, on average, the BO algorithm under this acquisition function has not yet converged to the global solution over the 120 iterations.

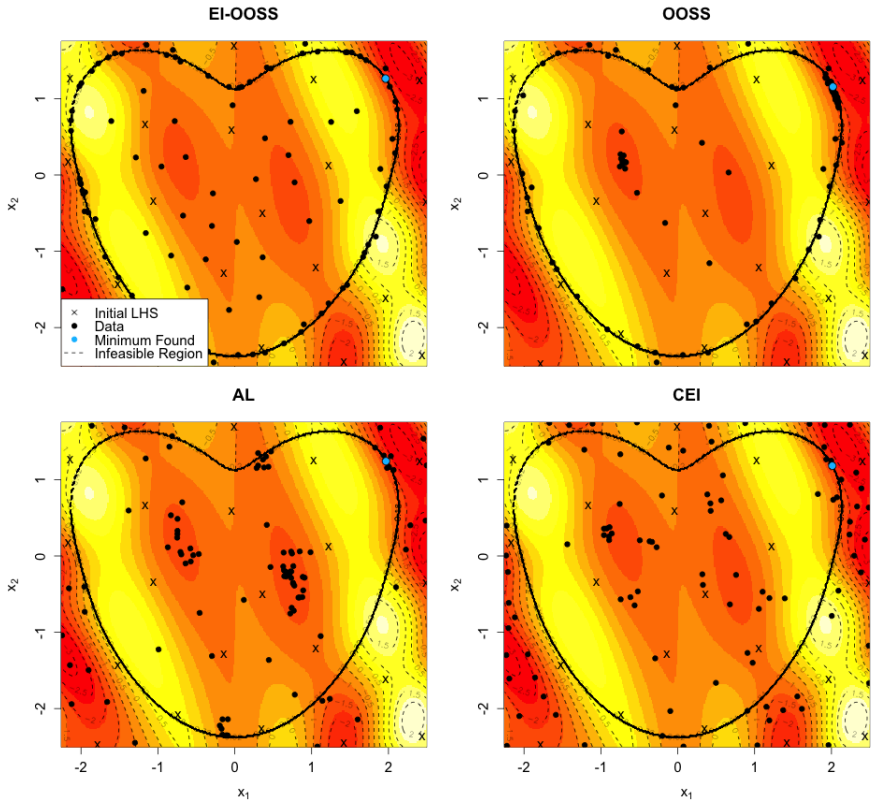


Figure 3: A view of the performance of the BO algorithm, using the four different acquisition functions, for a single run of the Monte Carlo experiment.

To better understand the behavior of the different acquisition functions, we take a look at a single run of the Monte Carlo experiment for each of the four different acquisition functions (Figure 3). Given the same initial LHD design to the four acquisition functions, we see very different behavior of the BO algorithms. We hypothesize that CEI did not fare as well as the other acquisition functions due to the fact that the algorithm tended to explore infeasible solutions quite often. Irrespective of feasibility, CEI favored exploring areas of the objective surface that were very low and avoided areas that were towards maximums. Likewise, the AL algorithm also explored the infeasible space but at a much lower rate as compared with CEI. However, within the feasible region the AL algorithm tended to explore the two local minima more often than the global minimum which may

explain the slower convergence on average to the global solution. On the other hand, both the EI-OOSS and OOSS acquisition functions spent the majority of their time exploring mainly the feasible space and the boundary of the feasible space, and do not explore far beyond the boundary, which allows more search efficiency. We note that while the original barrier methods were designed to never cross the feasibility boundary, our hybrid approach does explore along the boundary, sometimes crossing it. This behavior occurs because we are estimating the location of the boundary, and the statistical model learns the boundary by sometimes going just beyond it. Interestingly, in this one Monte Carlo simulation, OOSS tended to spend a brief amount of time searching one of the local minimums before jumping to the boundary where the global minimum was found. The behavior of the OOSS acquisition function occurs because early in the optimization, uncertainty is high, and so the barrier will be thick. In the Townsend problem, the global minimum lies along the feasible boundary at a steep edge and so a thick barrier will not allow any of the space near this solution to be reached. It is only in later iterations that the uncertainty decreases (and thus the barrier thickness decreases) enough that the optimization is able to start pushing against that part of the barrier and to reach the global solution. On the other hand, the effect of the improvement function in the EI-OOSS acquisition seemed to lead to an algorithm that explored the feasible space in a more evenly distributed (space-filling) fashion.

Lastly, we calculated the the average percent (over the 30 Monte Carlo runs) of infeasible points selected after initialization for each of the acquisition functions. The average percent of infeasible points selected was 20.7%, 30.0%, 29.9%, and 50.8% for the OOSS, EI-OOSS, AL, and CEI acquisition functions, respectively. Thus, OOSS resulted in the fewest infeasible evaluations of all of the algorithms.

4.2 Gramacy et al. 2016 Problem

Originally introduced in Gramacy et al. (2016), the optimization problem is a toy example with known solution and known comparators (Gramacy et al., 2016; Picheny et al., 2016). The problem contains a simple known linear function, and two unknown nonlinear

constraints. More formally, we state the problem as follows:

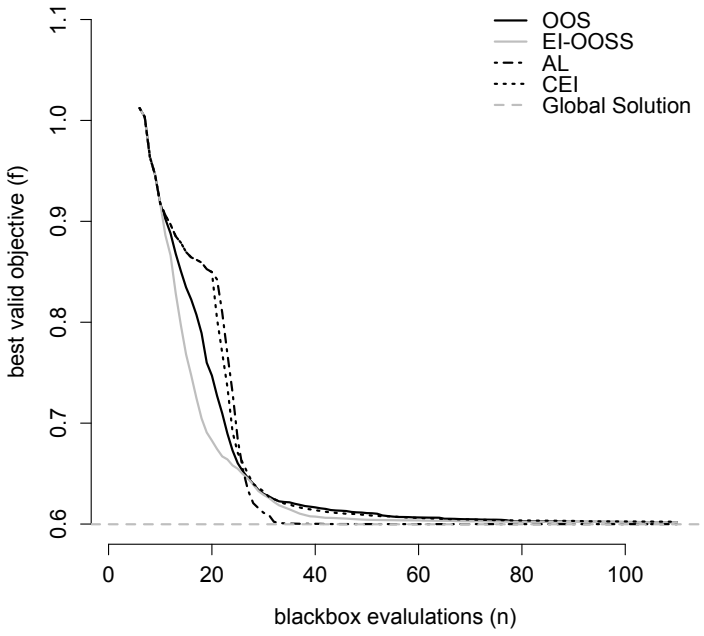
$$\begin{aligned} \min \ & f(x_1, x_2) = x_1 + x_2 \\ \text{s.t. } & c_1(x_1, x_2) = \frac{3}{2} - x_1 - 2x_2 - \frac{1}{2}\sin(2\pi(x_1^2 - 2x_2)), \\ & c_2(x_1, x_2) = x_1^2 + x_2^2 - \frac{3}{2} \end{aligned} \tag{20}$$

where the optimal solution is $f(x_1, x_2) = 0.5998$, which occurs along the constraint boundary at $(x_1, x_2) = (0.1954, 0.4044)$. We stress the fact that the objective function is a known function because Gramacy et al. (2016) treats the objective function as known rather than a black-box function. For sake of comparison we could also take this approach, however, in this example we choose to treat the objective function as a black-box function merely for illustration, as we postulate that having to model and quantify the uncertainty for the objective function (even as simple of a function as it is) should put our BO algorithm at a slight disadvantage as opposed to treating it as known. Mimicking the sample size conditions put forth in Gramacy et al. (2016), we start with an initial LHD of size 10 from the input space, and then sequentially select an additional 100 inputs to evaluate, and we repeat this Monte Carlo experiment a total of 100 times.

As seen in Figure 4, on average, all five algorithms converge to the global solution by around 60 iterations. AL tended to be slower at decreasing the best objective function values at the beginning, but then dramatically approached the global solution much faster than any other algorithm. On the other hand, EI-OOSS and OOSS were fast to decrease the objective function but then slowed down as compared to AL. At around 25 iterations, on average, OOSS and EI-OOSS seemed to converge to the global solution at the same rate as CEI. Again, the OOSS acquisition function selected, on average, the fewest number of infeasible points at 12.15%, followed by EI-OOSS, AL, and CEI at 30.7%, 63.65%, and 81.5%, respectively.

4.3 Pump-and-treat Hydrology Problem

A real-world hydrology computer experiment, the pump-and-treat hydrology problem (Matott et al., 2011) is based on a groundwater contamination scenario stemming from the Lockwood Solvent Groundwater Plume Site located near Billings, Montana. Years of in-



n	25	50	100
	95%		
OOSS	0.803	0.755	0.606
EI-OOSS	0.778	0.613	0.605
AL	1.023	0.601	0.600
CEI	0.831	0.624	0.608
	average		
OOSS	0.660	0.611	0.602
EI-OOSS	0.655	0.604	0.602
AL	0.684	0.600	0.599
CEI	0.671	0.609	0.603
	5%		
OOSS	0.604	0.601	0.600
EI-OOSS	0.602	0.600	0.600
AL	0.604	0.600	0.599
CEI	0.606	0.601	0.600

Figure 4: The results of running 100 Monte Carlo repetitions with random starting inputs. The plot and table show the average best valid objective function values found over 100 black-box iterations. 5^{th} and 95^{th} percentiles are also included to better understand the spread of the distribution on the Monte Carlo repetitions. Here the horizontal dashed line corresponds to the global solution (i.e., minimum) of the optimization problem.

dustrial practices have led to the formation of two plumes of chlorinated contaminants in the area that are slowly, and dangerously, migrating towards the Yellowstone river. Preventing the two plumes from reaching the Yellowstone river is of utmost importance to ensure the safety of the local water supplies. In order to stop the migration of the two plumes, a pump-and-treat remediation is proposed. Six pump-and-treat wells will be placed at the site of the plumes and these wells will then pump out the contaminated water from the soil, purify it, and then return the clean treated water to the soil. To better understand the dynamics of the physical system, and to come up with an optimal strategy, a computer

simulator was constructed to model the physical process. Here the inputs to the computer simulator are the pumping rates that can be set for the six pump-and-treat wells, and the output of the computer simulator is the cost associated with running the pump-and-treat wells and whether or not the containment of the two contaminated plumes was successful. Thus, the goal of the pump-and-treat hydrology problem is to minimize the cost of running the pump-and-treat wells while ensuring that the two contaminated plumes are contained.

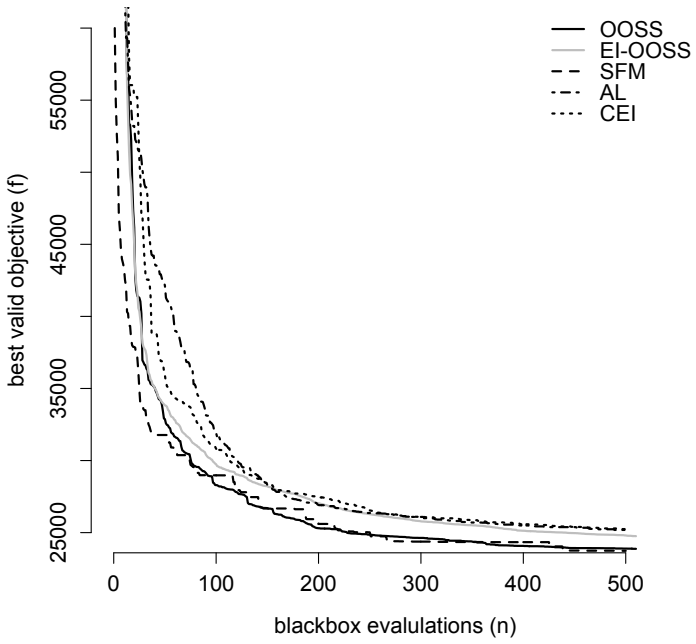
Casting the pump-and-treat hydrology problem in the framework of a constrained optimization, we formulate the problem as follows:

$$\min_x \{f(x) = \sum_{j=1}^{6} x_j : c_1(x) \leq 0, c_2(x) \leq 0, x \in [0, 20 \cdot 10^4]^6\}. \tag{21}$$

Here the objective function, f, is (known) linear and describes the cost associated with running the pump-and-treat wells. The two plumes of contaminants are contained when the two constraints, c_1 and c_2, are satisfied. The inputs $x_1, ..., x_6$ represent the six pumping rates that can be set for the six pump-and-treat wells within the computer simulator. The computer simulator is essentially a black-box function since, for any input configuration evaluated by the simulator, the only information that is returned is that of the objective and constraint values. Likewise, each input evaluation is an expensive one, and so the time it takes to run the computer simulator is nontrivial.

The pump-and-treat hydrology problem was solved (amongst other older poorer solutions) in Gramacy et al. (2016) using AL and in Pourmohamad and Lee (2020) using the statistical filter method (SFM), so we benchmark the results of our BO algorithm against theirs. Once again, we try to mimic the conditions put forth in those papers as closely as we can so that as fair of a comparison as possible can be made. Mimicking Gramacy et al. (2016), we start with an initial LHD of size 10 from the input space, and then sequentially select an additional 500 inputs to evaluate. Likewise, we repeat this Monte Carlo experiment a total of 30 times. Results are shown in Figure 5.

The BO algorithm seemed to perform best under the OOSS acquisition function, which was the only acquisition function able to challenge the SFM, dominating it through several stretches of iterations, and arriving at nearly the same best overall average value found. Overall, the OOSS and EI-OOSS acquisition functions were successful at minimizing the objective function and were highly competitive compared to AL, CEI and the SFM. All of



n	100	200	500
	95%		
OOSS	34332	30651	24155
EI-OOSS	35254	30824	25994
SFM	34763	30220	24742
AL	37040	29266	26050
CEI	34347	29553	25937
	average		
OOSS	28297	25305	23892
EI-OOSS	29738	27044	24804
SFM	28974	25604	23738
AL	31902	26912	25186
CEI	30814	27441	25246
	5%		
OOSS	24847	23881	23437
EI-OOSS	25996	24222	23717
SFM	27647	24464	23236
AL	26966	25191	24211
CEI	27692	26189	24512

Figure 5: The results of running 30 Monte Carlo repetitions with random starting inputs. The plot and table show the average best valid objective function values found over 500 black-box iterations. 5^{th} and 95^{th} percentiles are also included to better understand the spread of the distribution on the Monte Carlo repetitions.

the algorithms selected, on average, a very high number of infeasible points after initialization. Here, the OOSS acquisition function still performed the best with 86.25% of points being infeasible. The EI-OOSS, AL, and CEI acquisition followed with 87.4%, 94.8%, and 95.4%, respectively.

5 Discussion

Constrained optimization is a challenging task when the functions of interest arise from expensive black-box systems. BO has been shown, many times over, to be an effective solution to problems of this nature. The success of BO algorithms are clearly tied to the acquisition function they use for effectively guiding the search. The novelty of the work presented in this article is in the development of a new and efficient acquisition function for BO of expensive black-box constrained optimizations problems. Deriving the novel acquisition function from the successful hybridization of statistical surrogate modeling with barrier methods leads to a powerful acquisition function that is able to leverage both the efficient local search properties of the numerical method and the global search properties of the statistical model. We demonstrated the success of our new BO algorithm on a suite of test problems and a real-world computer experiment.

Our approach does require a choice of acquisition function for which we have provided two good candidate choices. Our OOSS acquisition function performed well in comparisons, and appeared to be the slightly better and more robust option. In some cases, careful tuning of γ in the EI-OOSS acquisition function may be capable of achieving slightly better results, but that does require rules for tuning, for which heuristical advice must be determined.

We speculate that barrier methods may perform well when the unconstrained objective function is complex near and beyond the boundary, as the barrier methods are more focused on keeping the search within the feasible region, while methods with less severe penalties may get more distracted exploring locally optimal values outside the feasible region. Indeed, with a sufficient penalty term, barrier methods will never evaluate a point that is infeasible in expectation. Our experiments have demonstrated that our approach evaluates fewer infeasible points than the comparators. This property may be particularly useful if a "safe BO" method is needed, such as when infeasible evaluations risk damaging equipment.

6 Supplementary Materials

Modified Townsend Code: `R` code for reproducing the Bayesian optimization algorithm used to solve the modified Townsend problem. (.R file)

Acknowledgments

We are grateful for the helpful comments and suggestions from two anonymous reviewers, an associate editor, and the editor.

Appendix

A Expectation of the log operator

Let x be a random variable. The expectation of $\log(x)$ can be approximated using a second order Taylor expansion of $\log(x)$ around $x_0 = \mathbb{E}(x)$, and then by taking the expectation of that Taylor expansion. A second order Taylor expansion around $x_0 = \mathbb{E}(x)$ yields

$$\log(x) \approx \log(\mathbb{E}(x)) + \frac{1}{\mathbb{E}(x)} \times (x - \mathbb{E}(x)) - \frac{1}{2}\frac{1}{\mathbb{E}(x)^2}(x - \mathbb{E}(x))^2. \tag{22}$$

Taking expectations we obtain

$$\mathbb{E}(\log(x)) \approx \mathbb{E}(\log(\mathbb{E}(x))) + \frac{1}{\mathbb{E}(x)} \times (\mathbb{E}(x) - \mathbb{E}(x)) - \frac{1}{2}\frac{1}{\mathbb{E}(x)^2}\mathbb{E}(x - \mathbb{E}(x))^2) \tag{23}$$

$$= \log(\mathbb{E}(x)) - \frac{1}{2}\frac{\mathbb{V}(x)}{\mathbb{E}(x)^2} \tag{24}$$

References

Brochu, E., Cora, V. M., and de Freitas, N. (2010). A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning.

Conn, A. R., Scheinberg, K., and Vicente, L. N. (2009). *Introduction to Derivative-Free Optimization*. SIAM, Philadelphia.

Eriksson, D., Pearce, M., Gardner, J., D., T. R., and Poloczek, M. (2019). Scalable global optimization via local Bayesian optimization. In *Advances in Neural Information Processing Systems*, pages 5497–5508.

Gardner, J. R., Kusner, M. J., Xu, Z., Weinberger, K. Q., and Cunningham, J. P. (2014). Bayesian optimization with inequality constraints. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '14, pages 937–945.

Gramacy, R. B. (2016). laGP: Large-scale spatial modeling via local approximate gaussian processes in R. *Journal of Statistical Software*, 72(1):1–46.

Gramacy, R. B. (2020). *Surrogates: Gaussian Process Modeling, Design, and Optimization for the Applied Sciences*. Chapman & Hall/CRC, Boca Raton, Florida, first edition.

Gramacy, R. B., Gray, G. A., Digabel, S. L., Lee, H. K. H., Ranjan, P., Wells, G., and Wild, S. M. (2016). Modeling an augmented lagrangian for blackbox constrained optimization. *Technometrics*, 58(1):1–11.

Jones, D., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black box functions. *Journal of Global Optimization*, 13:455–492.

Kleijnen, J. P. C. (2015). *Design and Analysis of Simulation Experiments*. Springer, New York, second edition.

Letham, B., Karrer, B., Ottoni, G., and Bakshy, E. (2019). Constrained bayesian optimization with noisy experiments. *Bayesian Analysis*, 14(2):495–519.

Lindberg, D. and Lee, H. K. H. (2015). Optimization under constraints by applying an asymmetric entropy measure. *Journal of Computational and Graphical Statistics*, 24:379–393.

Matott, L. S., Leung, K., and Sim, J. (2011). Application of matlab and python optimizers to two case studies involving groundwater flow and contaminant transport modeling. *Computers and Geosciences*, 37(1894–1899).

McKay, M. D., Conover, W. J., and Beckman, R. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21:239–245.

Mockus, J., Tiesis, V., and Zilinskas, A. (1978). The application of bayesian methods for seeking the extrenum. *Towards Global Optimization*, 2:117–129.

Nocedal, J. and Wright, S. J. (2006). *Numerical Optimization*. Springer, New York, NY, second edition.

Picheny, V., Gramacy, R. B., Wild, S., and Le Digabel, S. (2016). Bayesian optimization under mixed constraints with a slack-variable augmented Lagrangian. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29*, pages 1435–1443. Curran Associates, Inc.

Pourmohamad, T. and Lee, H. K. H. (2016). Multivariate stochastic process models for correlated responses of mixed type. *Bayesian Anal.*, 11(3):797–820.

Pourmohamad, T. and Lee, H. K. H. (2020). The statistical filter approach to constrained optimization. *Technometrics*, 62(3):303–312.

Santner, T. J., Williams, B. J., and Notz, W. I. (2003). *The Design and Analysis of Computer Experiments*. Springer-Verlag, New York, NY.

Schonlau, M., Jones, D., and Welch, W. (1998). Global versus local search in constrained optimization of computer models. In *New Developments and applications in experimental design*, number 34 in IMS Lecture Notes - Monograph Series, pages 11–25.

Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 25*, pages 2951–2959. Curran Associates, Inc.

Taddy, M., Lee, H. K. H., Gray, G. A., and Griffin, J. D. (2009). Bayesian guided pattern search for robust local optimization. *Technometrics*, 51:389–401.

Teh, Y. W., Newman, D., and Welling, M. (2007). A collapsed variational bayesian inference algorithm for latent dirichlet allocation. In Schölkopf, B., Platt, J. C., and Hoffman, T., editors, *Advances in Neural Information Processing Systems 19*, pages 1353–1360. MIT Press.

Townsend, A. (2014). Constrained optimization in Chebfun. https://www.chebfun.org/. Retrieved 2017-08-29.