Bayesian Optimization via Barrier Functions

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

May 3, 2020

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 paper, 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 ensuring 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.

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 few steps, 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). An appropriate acquisition function should accurately encode the beliefs about which is the next best 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.

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 (2019) 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, that is 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 ensuring that the boundary of the constraint space is never crossed. Our new BO approach is highly 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. Three 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 about potential future work and concluding remarks.

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 (Sacks et al., 1989; 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

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

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

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

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 defer further discussions of acquisitions to Section 3, but clearly different choices of acquisition functions should lead to different measures of belief of the search algorithm when searching for the next best input to evaluate. 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 (Lee et al., 2011; Gramacy et al., 2016; Letham et al., 2018).

3 Novel Acquisition Functions

At the heart of all Bayesian optimization algorithms is an acquisition function, $a(x)$, for effectively guiding the search. Bayesian optimization essentially embeds an 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. Furthermore, given that the GP surrogate model is essential for maximizing the acquisition function, it is also of tantamount importance 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 three 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 $y^{(i)} = BM(x^{(i)}; \gamma)$ 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 although the use of correlated surrogate models for f and c may yield improvements (Pourmohamad and Lee, 2016), we found that using independent 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 $y^{(i)} = BM(x^{(i)}; \gamma)$ with the following surrogate model

$$Y(x) = Y_f(x) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \log(-Y_{c_i}(x)). \tag{6}$$

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.,

$$\begin{aligned}
\min_x \mathbb{E}(Y(x)) &= \min_x \mathbb{E}\left(Y_f(x) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \log(-Y_{c_i}(x))\right) \\
&= \min_x \mathbb{E}\left(Y_f(x)\right) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \mathbb{E}\left(\log(-Y_{c_i}(x))\right) \\
&\approx \min_x \mathbb{E}\left(Y_f(x)\right) - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \left(\log(\mathbb{E}(-Y_{c_i}(x))) - \frac{\mathbb{V}(-Y_{c_i}(x))}{2\mathbb{E}(-Y_{c_i}(x))^2}\right) \\
&= \min_x \mu_f - \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \left(\log(-\mu_{c_i}) + \frac{\sigma_{c_i}^2}{2\mu_{c_i}^2}\right)
\end{aligned} \tag{7}$$

The derivation of the expectation of the log operator, in the third line of (7), is taken from Teh et al. (2007) and is a direct consequence of taking a second order Taylor expansion about $\mathbb{E}(-c_i(x))$. Now, it is clear to see that minimizing the predictive mean in (7) can be viewed as maximizing the following acquisition function:

$$a(x) = -\mu_f + \left(\frac{1}{\gamma}\right) \sum_{i=1}^{m} \left(\log(-\mu_{c_i}) + \frac{\sigma_{c_i}^2}{2\mu_{c_i}^2}\right). \tag{8}$$

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 (8) 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 subsequent sections, care must be taken when choosing the value of γ as it will be seen that γ plays a critical role in the exploration-exploitation tradeoff of our optimization algorithm. 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 (8) to measure our prediction uncertainty for the objective function, our acquisition function will be overly optimistic in exploring the objective function and will settle more often that not on 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, it is sufficient to set γ to be a "large" number to ensure local convergence of the barrier method algorithm. However in our case extra care must be taken in setting γ as it will be shown that the value of γ will play a critical role in balancing the exploration-exploitation tradeoff of our acquisition function. Recall the optimization problem in (4). When the constraints are met, we have that the quantity $\sum_{i=1}^m \log(-c_i(x)) > 0$. Now, when γ is large (i.e., tending to ∞) the quantity $1/\gamma \sum_{i=1}^m \log(-c_i(x))$ goes to 0 and results in a local (greedy) search algorithm. On the other hand, when γ is small (i.e., tending to 0) the quantity $1/\gamma \sum_{i=1}^m \log(-c_i(x))$ is positive and large, and results in a global search algorithm. From this point-of-view, γ can be seen as a tuning parameter that can be adjusted to control the level of search of the BO algorithm. One potential option for setting the value of γ would be to select an appropriate fixed value based on the complexity of the optimization problem and that would be at the discretion of subjectivity. We refer to this as the "fixed γ" acquisition function, or "fixed" for short. Although simple to implement, great care would need to be taken when choosing a fixed γ as an inappropriate choice might bias the search into too much exploration, or too little. A perhaps better choice would be to take an objective approach and allow 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 $\gamma = 1/\sigma_f^2$, where σ_f^2 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. When σ_f^2 is large there is a lot of uncertainty in our prediction of the objective function, and consequently γ will be small, leading to global search. Conversely, when σ_f^2 is small there is less uncertainty in our prediction of the objective function, and so γ will be large and lead to local search. Under this choice of γ we obtain the updated acquisition function

$$a(x) = -\mu_f + \sigma_f^2 \sum_{i=1}^{m} \left(\log(-\mu_{c_i}) + \frac{\sigma_{c_i}^2}{2\mu_{c_i}^2} \right). \tag{9}$$

We refer to this acquisition function as "One Over Sigma" (OOS), and it incorporates the uncertainty in both the objective and constraint functions and directly balances the exploration-exploitation based on this uncertainty. This acquisition function provides a robust balance between performance and elimination of the need to tune γ.

3.3 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 $I(x) = \max_x\{0, f_{\min}^n - Y(x)\}$ where, after n runs of the computer model, $f_{\min}^n = \min\{f(x_1), ..., f(x_n)\}$ is the current minimum value observed. 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.,

$$x^* = \arg\max_{x \in \mathcal{X}} \mathbb{E}\{I(x)\}. \tag{10}$$

Fortunately, 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

$$\mathbb{E}(I(x)) = (f_{\min}^n - \mu^n(x))\Phi\left(\frac{f_{\min}^n - \mu^n(x)}{\sigma^n(x)}\right) + \sigma^n(x)\phi\left(\frac{f_{\min}^n - \mu^n(x)}{\sigma^n(x)}\right) \tag{11}$$

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 (11) 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 ($\sigma(x)$).

Now, although the EI acquisition function was originally developed for the case of unconstrained optimization, we can exploit its natural exploration-exploitation characteristics by inserting it into the minimization problem in (7). Replacing the objective function's surrogate model, $Y_f(x)$, with the improvement function $-I(x)$ in (6), yields

$$\min_x \mathbb{E}\left(-I(x) - \left(\frac{1}{\gamma}\right)\sum_{i=1}^m \log(-c_i(x))\right) = \min_x -\mathbb{E}(I(x)) - \left(\frac{1}{\gamma}\right)\sum_{i=1}^m \left(\log(-\mu_{c_i}) + \frac{\sigma_{c_i}^2}{2\mu_{c_i}^2}\right). \tag{12}$$

Note that since we are minimizing in (7) we will need to use the negative improvement function. The minimization problem in (12) leads to the following acquisition function

$$a(x) = (f_{\min}^n - \mu_f)\Phi\left(\frac{f_{\min}^n - \mu_f}{\sigma_f}\right) + \sigma_f\phi\left(\frac{f_{\min}^n - \mu_f}{\sigma_f}\right) + \left(\frac{1}{\gamma}\right)\sum_{i=1}^m \left(\log(-\mu_{c_i}) + \frac{\sigma_{c_i}^2}{2\mu_{c_i}^2}\right). \tag{13}$$

The parameter γ is a fixed constant in (13), however, present now in the acquisition function is a variance term for the objective function f which will help our BO algorithm in the tradeoff between exploring locally versus globally. The EI acquisition function shows great promise when γ can be tuned optimally.

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, 2019), 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. In the case where comparator solutions existed, we tried to mimic the conditions of the comparator algorithms as to facilitate a fair comparison amongst algorithms. As well, we solved each of the three problems using all three variations of the proposed acquisition function in order to compare and contrast them. For γ, we consider values from 10 to $1,000,000$. Lastly, there are many tools and software packages available for fitting GPs to data, however, for all of our examples we favor using 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 (14) 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:

$$\begin{aligned} \min \;\; & f(x_1, x_2) = \sin(x_2)\exp\{(1-\cos(x_1))^2\} + \cos(x_1)\exp\{(1-\sin(x_2))^2\} + (x_1 - x_2)^2 \\ \text{s.t.} \;\; & c(x_1, x_2) = x_1^2 + x_2^2 - \left(2\cos(t) - \frac{1}{2}\cos(2t) - \frac{1}{4}\cos(3t) - \frac{1}{8}\cos(4t)\right)^2 - (2\sin(t))^2 \end{aligned} \tag{14}$$

where $t = \text{arctan2}(x_1, x_2)$, $-2.25 \leq x_1 \leq 2.5$, 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.

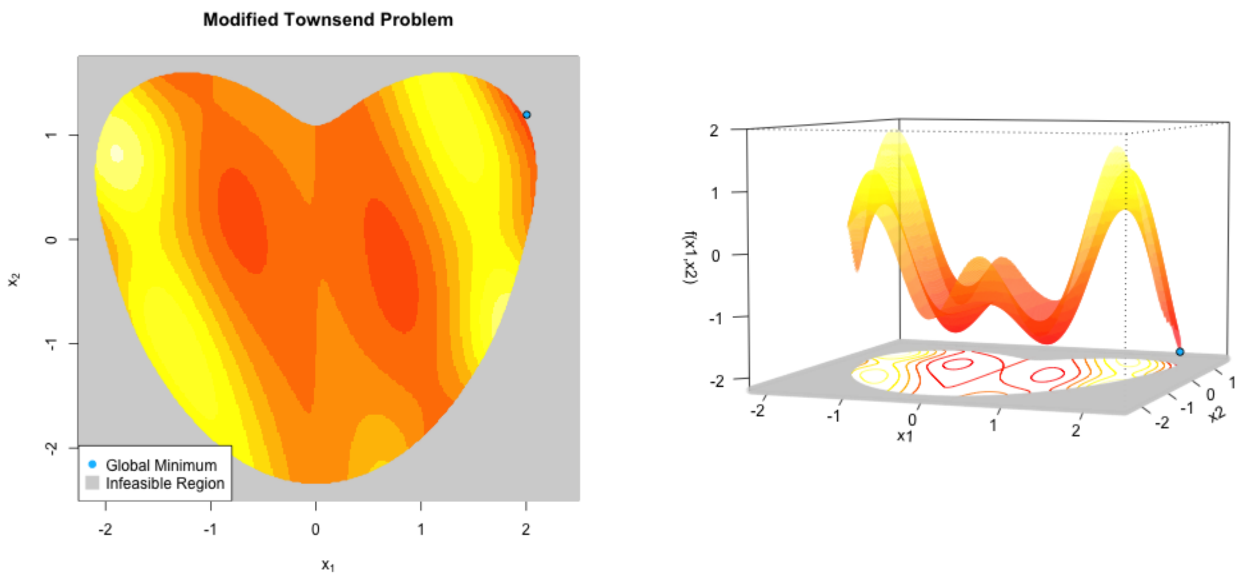
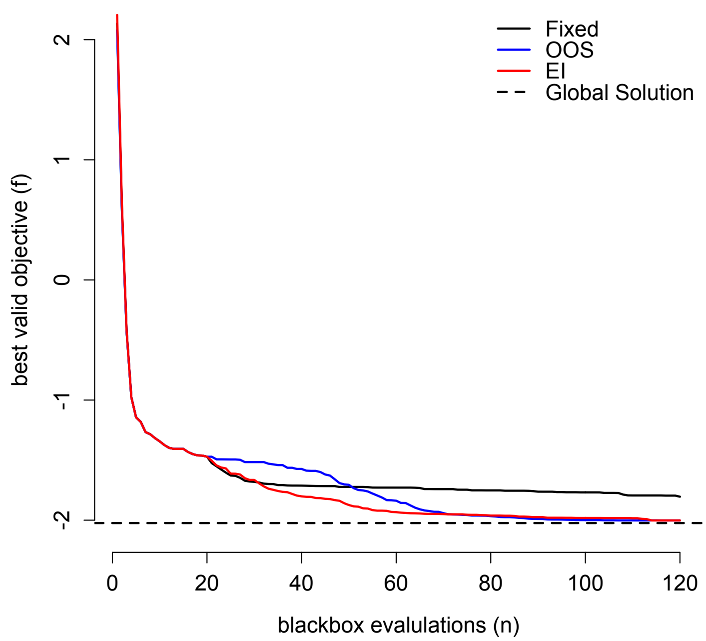


Figure 1: A view of the the objective function of the modified Townsend problem subject to the constraint function. The problem contains several local minima, and the global minimum is known to lie along the boundary of the feasible space.

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 our novel acquisition function to guide the search. As a general guideline, we follow the rule of thumb put forth by Loeppky et al. (2009), and require that the number of initial inputs be about ten times the input dimension, that is, n = 10d, to achieve reasonable GP surrogate model fits for f and c.

We solve the modified Townsend problem using the the three variations of acquisitions functions found in Section 3. Here we denote the three acquisitions functions as either fixed, one over σ^2 (OOS), or expected improvement (EI), based on sections 3.1, 3.2, and

3.3, respectively. For each of the three acquisitions 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 we apply the same initial LHD, of size 20, to each of the acquisition functions during a given Monte Carlo experiment. We set $\gamma = 1,000,000$ for both the fixed and EI acquisition functions.



n	45	70	120
	95%		
Fixed	-1.638	-1.639	-1.640
OOS	-1.352	-1.831	-1.963
EI	-1.625	-1.659	-1.954
	average		
Fixed	-1.716	-1.741	-1.805
OOS	-1.620	-1.942	-2.003
EI	-1.822	-1.949	-2.002
	5%		
Fixed	-2.007	-2.017	-2.017
OOS	-1.942	-2.018	-2.021
EI	-2.007	-2.020	-2.022

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.

On average, both the OOS and EI acquisition functions were able to find the global solution of the problem over the additional 100 input-output updates (Figure 2), with EI being much quicker at decreasing the objective function than OOS at the early stages of the BO algorithm. In fact, the fixed acquisition function does a better job at minimizing the problem at the early stages of the BO algorithm as compared to OOS as well. However, the OOS acquisition function does steadily decrease the objective function in the search for the global minimum and, at around 60 iterations, has caught up to the EI acquisition function and in the end has consistently found the global solution of the problem. The same cannot be said about the fixed acquisition function as, on average, the BO algorithm under this fixed acquisition function has not yet converged to the global solution. Looking at the lower 5% quantile of the table in Figure 2 we see that the BO algorithm does indeed approach the solution of the modified Townsend problem under the fixed acquisition function, and so, for some of the Monte Carlo repetitions the BO algorithm is finding the solution under this acquisition function. We do not believe that it is the case that, for the fixed acquisition function, the algorithm need be run longer to find the global solution, but that the algorithm is getting stuck in local minima due to the lack of a variance term for the objective function in the fixed acquisition function. To better understand this, we take a look at a single run of the Monte Carlo experiment for each of the three different acquisition functions (Figure 3).

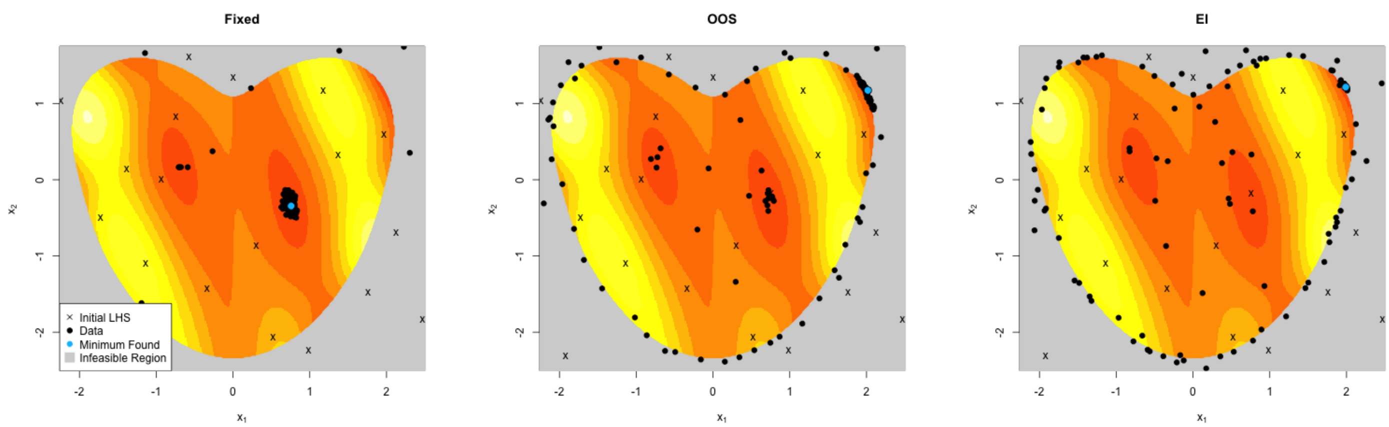


Figure 3: A view of the performance of the BO algorithm, using the three different versions of the acquisition function, for a single run of the Monte Carlo experiment.

Given the same initial LHD design to the three acquisition functions, we see very different behavior of the BO algorithm. As hypothesized, the fixed acquisition function was quick to get stuck in a local minima and so it did not explore the input space well. The fixed acquisition function suffers from the lack of a variance term for the objective function, and so extra care must be taken when choosing an appropriate value of γ. Choosing γ small promotes more global search, so with a small value for γ, say $\gamma = 10$, we see that the BO algorithm, under the fixed acquisition function, does a much better job of solving the modified Townsend problem (Figure 4) as opposed to a much larger value of γ. On the other hand, the OOS and EI acquisition functions spend their time exploring the space much more globally. Interestingly, in this instance of the algorithm, it would seem that the EI acquisition function tended not to get stuck in any of the local minima whereas the OOS acquisition function spent some partial amount of time investigating the local minima before deciding to search elsewhere globally.

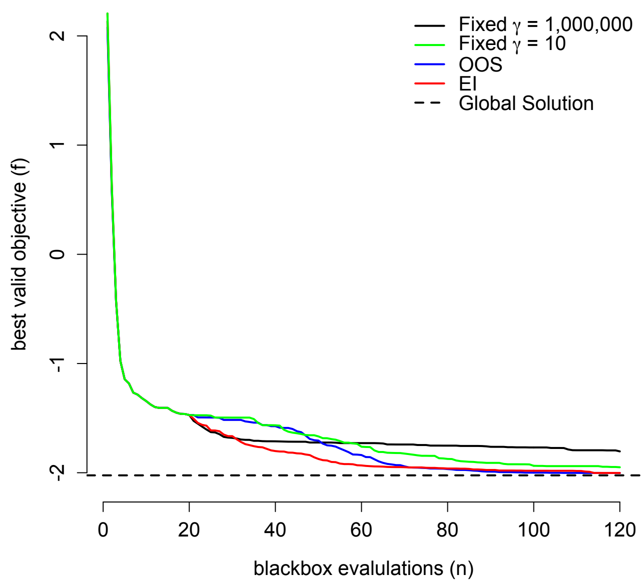


Figure 4: The average best valid objective function values found over 120 black-box iterations for 30 Monte Carlo repetitions with random starting inputs. The effect of making γ smaller for the fixed acquisition function leads to better performance of the BO algorithm.

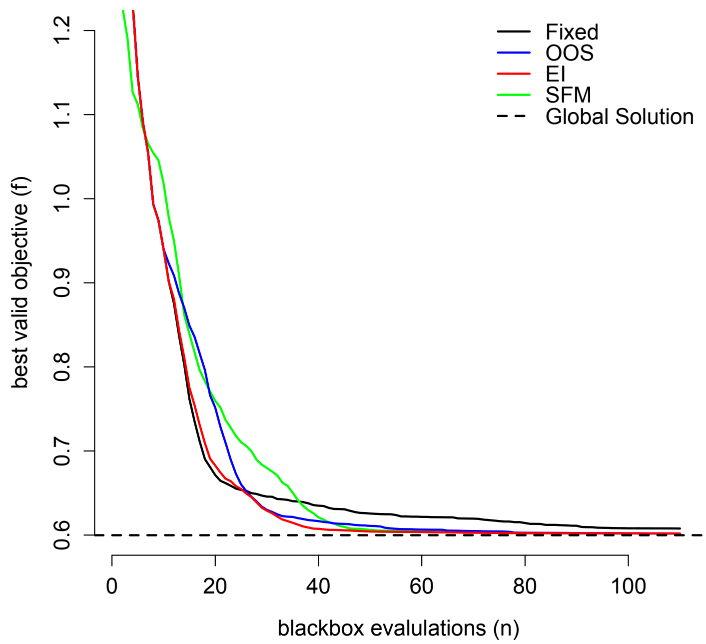
4.2 Gramacy et. al 2016 Problem

Originally introduced in Gramacy et al. (2016), the optimization problem is a toy example with known solution and known comparators (Gramacy et al., 2016; Pourmohamad and Lee, 2019). The problem contains a simple known linear function, and two unknown nonlinear constraints. More formally, we state the problem as follows:

$$\begin{aligned}
\min \ & f(x_1, x_2) = x_1 + x_2 \\
\text{s.t. } & c_1(x_1, x_2) = \frac{3}{2} - x_1 - 2x_2 - \frac{1}{2}\sin(2\pi(x_1^2 - 2x_2)), \\
& c_2(x_1, x_2) = x_1^2 + x_2^2 - \frac{3}{2}
\end{aligned} \tag{15}$$

where the optimal solution is $f(x_1, x_2) = 0.5998$, which occurs along the constraint space boundary at $(x_1, x_2) = (0.1954, 0.4044)$. We stress the fact that the objective function is a known function because in the works of Gramacy et al. (2016) and Pourmohamad and Lee (2019) they do not treat the objective function as a black-box function but rather as a function whose analytical form is known. 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 the case where we treat it as known. Mimicking the sample size conditions put forth in Gramacy et al. (2016) and Pourmohamad and Lee (2019), we start with an initial LHD of size 10 from the input space, and then sequentially select an additional 100 inputs to evaluate. Likewise, following Gramacy et al. (2016) and Pourmohamad and Lee (2019), we repeat this Monte Carlo experiment a total of 100 times. We set $\gamma = 1,000,000$ for both the fixed and EI acquisition functions.

As seen in Figure 5, on average, the three acquisition functions exhibit somewhat similar behavior as seen before in the modified Townsend problem. The OOS acquisition function tends to lag behind the other two acquisition functions, but eventually the OOS acquisition function passes the fixed acquisition functions, and then catches up to the EI acquisition for the remainder of the search. However, this time the fixed acquisition function with $\gamma = 1,000,000$ did not seem to get stuck in a local minima but did still tend to be slower to find the global minimum of the problem. However, in the end all three of the acquisition functions are able to converge towards the global solution to the problem. Pourmohamad and Lee (2019) solved this minimization problem in (15) using a statistical filter method (SFM) approach, and showed their method to have superior performance in comparison to the augmented Lagrangian method proposed in Gramacy et al. (2016). Due to this fact, we compare our BO algorithm to that of the results presented in Pourmohamad and Lee



n	25	50	100
	95%		
Fixed	0.773	0.760	0.754
OOS	0.803	0.755	0.606
EI	0.778	0.613	0.605
SFM	0.769	0.616	0.604
	average		
Fixed	0.654	0.626	0.608
OOS	0.660	0.611	0.602
EI	0.655	0.604	0.602
SFM	0.710	0.606	0.600
	5%		
Fixed	0.601	0.600	0.600
OOS	0.604	0.601	0.600
EI	0.602	0.600	0.600
SFM	0.606	0.600	0.599

Figure 5: 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 blackbox iterations. 5^{th} and 95^{th} percentiles are also included to better understand the spread of the distribution on the Monte Carlo repetitions.

(2019). As we can see from Figure 5, for the first approximately 30 sequential updates our BO algorithm does a much better job at finding the best current minimum for all three acquisition functions. In fact, with the exception of the fixed acquisition function, the OOS and EI acquisition functions do a much better job of fast reliable convergence towards the global minimum than the statistical filter method.

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 industrial practices have led to the formation of two plumes of chlorinated contaminants in the area, and these two contaminated plumes are slowly, and dangerously, migrating towards the Yellowstone river. Preventing the two plumes from reaching the Yellowstone river is of utmost importance as it helps 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 (two pump-and-treat wells will be placed at one of the plume sites while the other plume site will contain four pump-and-treat wells) and these pump-and-treat 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 under study. 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:

$$\min_x \{f(x) = \sum_{j=1}^{6} x_j : c_1(x) \leq 0, c_2(x) \leq 0, x \in [0, 20 \cdot 10^4]^6\}. \tag{16}$$

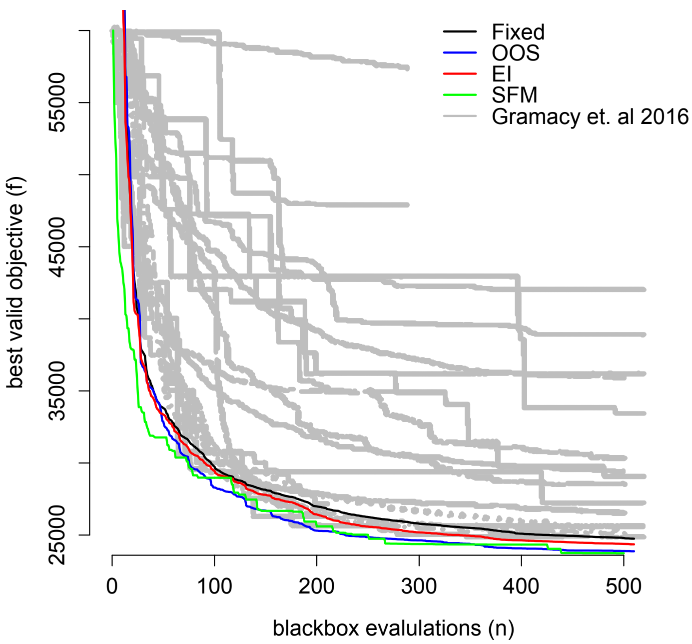
Here the objective function, f, is (known) linear and describes the cost associated with running the pump-and-treat wells. The two plumes of contaminants are contained when the two constraints, c_1 and c_2, are satisfied. The inputs $x_1, ..., x_6$ represent the six pumping rates that can be set for 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 in both Gramacy et al. (2016) and Pourmohamad and Lee (2019) (amongst other older poorer solutions) and 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. To encourage slightly more global exploration, we set $\gamma = 10$ for both the fixed and EI acquisition functions.The results in Pourmohamad and Lee (2019) were shown to be superior than that of Gramacy et al. (2016), and so in the table in Figure 6 we only include the statistical filter method (SFM) as a comparator, however, in Figure 6 we overlay our results on top of all of the results from Gramacy et al. (2016) and Pourmohamad and Lee (2019).

The BO algorithm seemed to perform best under the OOS acquisition function, with EI acquisition function being only slightly better than the fixed acquisition function. The BO algorithm, under all of the acquisition functions, did appear to be as good, if not better, than the methods examined in Gramacy et al. (2016), with the OOS acquisition function clearly outperforming all of the methods presented in Gramacy et al. (2016). On the other hand, only the OOS acquisition function was 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 BO algorithm was successful at minimizing the objective function and was highly competitive, if not better, with the methods of Gramacy et al. (2016) and Pourmohamad and Lee (2019).

5 Discussion

Constrained optimization is a challenging task when the functions of interests 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



n	100	200	500
	95%		
Fixed	35005	30824	25841
OOS	34332	30651	24155
EI	35188	30787	24951
SFM	34763	30220	24742
	average		
Fixed	29748	26990	24796
OOS	28297	25305	23892
EI	29462	26435	24384
SFM	28974	25604	23738
	5%		
Fixed	25824	24156	23717
OOS	24847	23881	23437
EI	26163	24075	23661
SFM	27647	24464	23236

Figure 6: 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.

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, and some of those choices contain a tuning parameter γ. Our OOS acquisition function performed well in comparisons, and does not require a tuning parameter, so it is the most robust and straightforward option. In some cases, careful tuning of γ in the EI acquisition function may achieve slightly better results, but that does require tuning, for which we have provided some heuristical advice.

Avenues for extensions to BO acquisition functions are endless and provide for further research. A potential extension of the fixed and EI acquisition functions could be to explore cooling schedules for γ, as is done in simulated annealing (Kirkpatrick et al., 1983). One could envision starting with a small value of γ (global search), and allowing for the value of γ to grow (local search) as a function of the sample size n as the search progresses.

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.

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, 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.

Kirkpatrick, S., Gelatt, C. D., and Vecci, M. P. (1983). Optimization by simulated annealing. *Science*, 220:671–680.

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

Lee, H. K. H., Gramacy, R. B., Linkletter, C., and Gray, G. A. (2011). Optimization subject to hidden constraints via statistical emulation. *Pacific Journal of Optimization*, 7:467–478.

Letham, B., Karrer, B., Ottoni, G., and Bakshy, E. (2018). Constrained bayesian optimization with noisy experiments. *Bayesian Anal.*

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.

Loeppky, J. L., Sacks, J., and Welch, W. J. (2009). Choosing the sample size of a computer experiment: A practical guide. *Technometrics*, 4:366–376.

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.

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. (2019). The statistical filter approach to constrained optimization. *Technometrics*, 0(0):1–10.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical Science*, 4:409–435.

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

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.