Constrained Efficient Global Optimization of Expensive Black-box Functions

Wenjie Xu [1 2] **Yuning Jiang** [1] **Bratislav Svetozarevic** [2] **Colin N. Jones** [1]

Abstract

We study the problem of constrained efficient global optimization, where both the objective and constraints are expensive black-box functions that can be learned with Gaussian processes. We propose CONFIG (**CON**strained ef**FI**cient **G**lobal Optimization), a simple and effective algorithm to solve it. Under certain regularity assumptions, we show that our algorithm enjoys the same cumulative regret bound as that in the unconstrained case and similar cumulative constraint violation upper bounds. For commonly used Mátern and Squared Exponential kernels, our bounds are sublinear and allow us to derive a convergence rate to the optimal solution of the original constrained problem. In addition, our method naturally provides a scheme to declare infeasibility when the original black-box optimization problem is infeasible. Numerical experiments on sampled instances from the Gaussian process, artificial numerical problems, and a black-box building controller tuning problem all demonstrate the competitive performance of our algorithm. Compared to the other state-of-the-art methods, our algorithm significantly improves the theoretical guarantees while achieving competitive empirical performance.

1. Introduction

Global optimization of expensive black-box functions is a pervasive problem in science and engineering. In black-box optimization problems, we typically need to evaluate different candidate solutions sequentially to find the optimal one without explicit gradient information. For example, hyperparameter tuning for machine learning models (Bergstra & Bengio, 2012; Snoek et al., 2015), control system performance optimization (Bansal et al., 2017; Xu et al., 2022b) and drug discovery or materials design (Negoescu et al., 2011; Frazier & Wang, 2016) can all be formulated as black-box optimization problems.

In these applications, we typically know little about the shape of the black-box functions, which can be non-convex and multi-modal. As a result, a global optimization method is needed. In many applications, e.g., tuning the hyperparameters of a deep neural network, each function evaluation can take a large number of resources such as energy, time, or GPU computation, so we want to use as few function evaluations as possible. Standard global optimization methods, such as genetic algorithms or simulated annealing (Pardalos & Romeijn, 2013), typically require a large number of samples and thus can not be applied to the optimization of expensive black-box functions. Therefore, Gaussian process based efficient global optimization[1](Jones et al., 1998; Shahriari et al., 2015), as a sample-efficient method to solve black-box optimization problems (Xu et al., 2022a), has recently been becoming more and more popular. The general idea of efficient global optimization is to build a surrogate model, typically by Gaussian process regression, of the black box function from samples and then to choose a search direction using this model. This problem, when formulated as a (kernelized) multi-armed bandit problem, is widely studied in the literatures (Audibert et al., 2010; Srinivas et al., 2012; Soare et al., 2014; Chowdhury & Gopalan, 2017a; Amani et al., 2019; Zhou & Ji, 2022).

A key challenge in many applications is black-box constraints. For example, when tuning the temperature controller parameters of a building, we need to minimize the energy consumption while keeping the occupants' comfort at user-defined levels, where both the energy and the comfort are black-box functions of the controller parameters. To handle the unknown black-box constraints, a variety of efficient global optimization methods with constraints have been developed. Intuitively, there is a non-trivial *tradeoff over optimization, learning and violations*. We can roughly classify the existing methods into different groups based on whether constraint violations are allowed during the optimization process. For the setting where no constraint violations are tolerated, a group of safe optimization meth-

[1]Automatic Control Laboratory, EPFL, Lausanne, Switzerland [2]Urban Energy Systems Laboratory, Empa, Zurich, Switzerland. Correspondence to: Yuning Jiang <yuning.jiang@ieee.org>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

[1]Also known as Gaussian process optimization (e.g., in (Srinivas et al., 2012)), Bayesian optimization (e.g., in (Frazier, 2018; Hernández-Lobato et al., 2016)) or kriging (e.g., in (Jeong et al., 2005)).

ods has been developed (Sui et al., 2015; Wu et al., 2016; Sui et al., 2018; Turchetta et al., 2020; Moradipari et al., 2020; Amani et al., 2020). However, these safe algorithms require a set of feasible solutions known a priori, while in practice, we may not get access to such feasible solutions or even do not know the feasibility of the problem. Furthermore, due to these hard safety-critical constraints, these algorithms may become stuck at a local minimum. The other extreme is where constraint violations are allowed to occur during the optimization process without penalty, for which there exists generic constrained Bayesian (efficient global) optimization methods (Schonlau, 1997; Sasena et al., 2002; Basudhar et al., 2012; Bagheri et al., 2017; Zhang et al., 2021). Along this line of research, a group of popular methods (Gardner et al., 2014; Gelbart et al., 2014) encodes the constraint information into the acquisition function (e.g., Constrained Expected Improvement). However, there are no theoretical guarantees on optimality, constraint violation incurred during optimization, or convergence rates for this class of methods. A recent work of (Inatsu et al., 2022) considers Bayesian optimization for distributionally robust chance-constrained problems but can not provide cumulative regret/violation bounds.

More recently, a third group of approaches has been studied (e.g., (Xu et al., 2022b; Zhou & Ji, 2022)), where constraint violations are allowed, but limits are placed on the total violation incurred during the optimization process. In this setting, two more recent works adopt a penalty-function approach (Lu & Paulson, 2022) and a primal-dual approach (Zhou & Ji, 2022) to solve constrained efficient global optimization problems. The common ideas behind these two approaches are the addition of a penalizing term of the constraint violation in the objective and the transformation of the constrained problem to an unconstrained one. However, both of these methods require choosing values for some critical parameters (e.g., penalty coefficient (Lu & Paulson, 2022) and dual update step size (Zhou & Ji, 2022)). The performance of these methods relies on the chosen parameters heavily, and hence these methods can take significant parameter tuning effort for implementation. Additionally, the cumulative violations considered in (Zhou & Ji, 2022) are the violations of the cumulative constraint values, not the real total violations. Such a form of cumulative violation bounds is weak and can not rule out the case where points with severe violations and small constraint values are alternatingly sampled while keeping a low cumulative constraint value. The work of (Lu & Paulson, 2022) only provides a penalty-based regret bound, which is defined as the regret plus a penalty weight times violations, which does not directly lead to separate bounds on regret and violation since the penalty weight is unknown beforehand. We also notice that a concurrent work (Guo et al., 2022) extends the fixed penalty to an increasing penalty sequence. However, it requires a delicate design of the penalty sequence and may result in numerical issues when the penalty is large. Last but not least, all the existing methods (Sui et al., 2015; Gelbart et al., 2014; Zhou & Ji, 2022; Lu & Paulson, 2022), either require an initial feasible solution to be given or lack the scheme to detect and report infeasibility when the original problem is infeasible. This is key, as, in practice, black-box optimization problems may be infeasible, and a scheme to detect and report infeasibility is desired.

Our contributions. We propose CONFIG, a simple and effective algorithm for the constrained efficient global optimization of expensive black-box functions by exploiting the principle of *optimism in the face of uncertainty*. Specifically,

1. CONFIG dynamically maintains an optimistic estimate of the feasible set as demonstrated in Fig. 1 and solves an auxiliary problem with lower confidence bound (LCB) surrogates as both objective and constraints to get the next sample in each step. Intuitively, CONFIG trades potential violations for faster learning of both the objective function and the constraints.

2. Under certain regularity assumptions, we show that the algorithm enjoys the same cumulative regret bound as that in the unconstrained case. In addition, we show that the cumulative constraint violation has an upper bound in terms of maximum information gain, which is similar to the cumulative regret bounds. For the commonly used Mátern and Squared Exponential kernels, our bounds are sublinear and allow us to derive a convergence rate to the optimal solution of the original constrained problem.

3. Furthermore, our method naturally provides a scheme to declare infeasibility when the original black-box optimization problem is infeasible.

4. Numerical results show that our algorithm achieves performance competitive with the state-of-the-art algorithms in terms of both optimality and constraint violations.

In sharp contrast, existing algorithms for constrained Bayesian optimization, including the popular CEI method (Gardner et al., 2014) and the recently introduced primal-dual method (Zhou & Ji, 2022; Xu et al., 2023), can only provide no or weak guarantees on the cumulative regret/violation bounds or on the convergence to the optimal solution (See Tab. 1 for details). Additionally, none of the existing methods can detect infeasibility if the original problem is infeasible. As such, our algorithm significantly improves the theoretical guarantees while achieving competitive empirical performance compared to the existing state-of-the-art constrained global/Bayesian optimization methods.

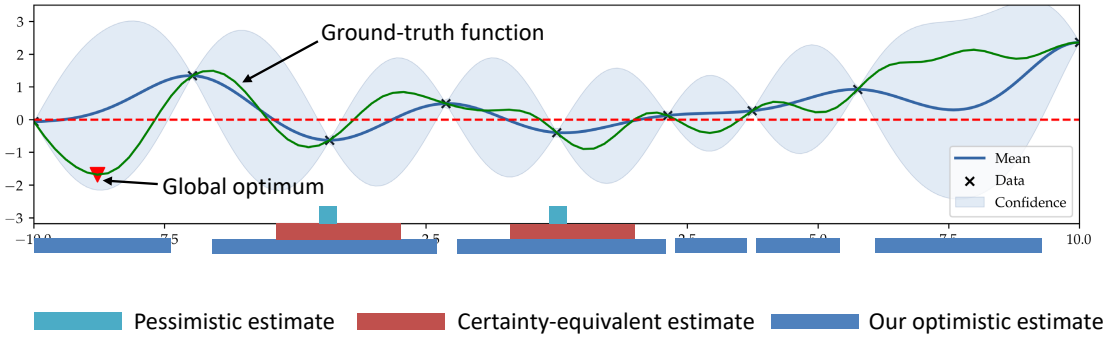

Figure 1. A simple demonstration of different ways to construct the estimate of the feasible set, where we minimize an unknown function subject to that it is non-positive. The pessimistic estimate can guarantee feasibility with high probability. However, it is tiny and can easily be empty if the initial points are infeasible. With the certainty-equivalent estimate, the algorithm may easily get stuck due to a lack of exploration. In contrast, our optimistic estimate of the feasible set contains the ground-truth feasible set with high probability.

Table 1. The comparison of our method to existing state-of-the-art constrained efficient global optimization (or kernelized multi-armed bandits with constraints) methods.

Works	**Cumulative Regret Bound**	**Cumulative Violation Bound**	**Optimality Guarantee**	**Infeasibility Declaration**
SafeOPT (Sui et al., 2015), etc.	✗	No violation	Local convergence	✗
Constrained EI (Gardner et al., 2014), etc.	✗	✗	✗	✗
Primal-Dual (Zhou & Ji, 2022), etc.	✓	weak-form	✗	✗
Penalty function method (Lu & Paulson, 2022), etc.	Penalty-based Regret Bound		Global convergence with penalty-dependent rate	✗
CONFIG (This paper)	✓	✓	Global convergence	✓

2. Notation and Preliminaries

This section introduces the class of constrained global optimization problems studied in this paper and recaps existing results from the field of efficient global optimization.

2.1. Problem Statement

We consider the following black-box optimization problem,

$$\min_{x \in \mathcal{X}} \quad f(x) \ , \tag{1a}$$

$$\text{subject to} \quad g_i(x) \leq 0 \ , \quad \forall i \in [N], \tag{1b}$$

where $\mathcal{X} \subset \mathbb{R}^d$ is a known set of candidate solutions (e.g., a hyperbox) with dimension d and $[N] := \{1, 2, \ldots, N\} \subset \mathbb{Z}$ denotes the set of integers from 1 to N. Both f and $g_i, i \in [N]$ are unknown black-box functions.

Remark 1. In Prob. 1, we only consider inequality constraints because an equality constraint $g(x) = 0$ can be equivalently transformed into two inequality constraints $g(x) \leq 0, -g(x) \leq 0$.

To proceed, we make the following commonly used assumptions in the field of efficient global optimization.

Assumption 2.1. Set $\mathcal{X}$ is compact.

Assumption 2.1 is naturally satisfied in a large set of applications. For example, when tuning the hyperparameters of a machine learning model, we can restrict the parameters to lie inside a hyperbox based on prior knowledge.

Assumption 2.2. $f \in \mathcal{H}_{k_0}$, $g_i \in \mathcal{H}_{k_i}, i \in [N]$, where $k_i : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}, i \in \{0\} \cup [N]$ are symmetric, positive-semidefinite kernel functions and $\mathcal{H}_{k_i}$ are their corresponding reproducing kernel Hilbert spaces (RKHSs, see (Schölkopf et al., 2001)). Furthermore, we assume $\|f\|_{k_0} \leq B_0$ and $\|g_i\|_{k_i} \leq B_i, i \in [N]$, where $\|\cdot\|_{k_i}$ is the norm induced by the inner product in the corresponding reproducing kernel Hilbert space.

Assumption 2.2 requires that the underlying functions are regular in the sense of having a bounded norm in an RKHS. In existing literature of efficient global optimization (also known as Bayesian optimization or Gaussian process optimization), having a bounded norm in an RKHS is a common assumption (e.g., (Srinivas et al., 2012; Zhou & Ji, 2022)).

Assumption 2.3. Prob. (1) is feasible and has an optimal solution x^* with the optimal value $f^* = f(x^*)$.

With Assumption 2.3, we focus on the case that the original black-box optimization problem is feasible. We will separately discuss how our algorithm can handle infeasible

problem instances later in Sec. 5.

We consider a zero-order feedback setting, where the algorithm can sequentially query noisy evaluations of different candidate solutions to learn the optimal feasible solution. At step t, with query point x_t, we get noisy evaluations,

$$y_{0,t} = f(x_t) + \xi_{0,t} \ , \tag{2a}$$
$$y_{i,t} = g_i(x_t) + \xi_{i,t} \ , \quad i \in [N], \tag{2b}$$

where $\xi_{i,t}, i \in \{0\} \cup [N], t \geq 1$ are i.i.d σ-sub-Gaussian noise with fixed $\sigma > 0$. We also use $X_t := (x_1, x_2, \cdots, x_t)$ to denote the sequence of sampled points up to step t.

2.2. Gaussian Process Surrogates

To design our algorithm, we use a Gaussian process surrogate model to learn the unknown functions. As in (Chowdhury & Gopalan, 2017a), we artificially introduce a Gaussian process $\mathcal{GP}(0, k(\cdot,\cdot))$ for the surrogate modeling of the unknown black-box function f. We also adopt an *i.i.d* Gaussian zero-mean noise model with noise variance $\lambda > 0$.

Remark 2. Note that this Gaussian process model is *only* used to derive posterior mean functions, covariance functions, and maximum information gain for the purpose of algorithm design. We introduce the artificial Gaussian process only to make the algorithm more interpretable. It does not change our set-up that f is a deterministic function and that the noise is only assumed to be sub-Gaussian. Indeed, we are considering the agnostic setting introduced in (Srinivas et al., 2012).

We introduce the following functions for x, x',

$$\mu_{0,t}(x) = k_0(x_{1:t}, x)^\top \left(K_{0,t} + \lambda I\right)^{-1} y_{0,1:t} \ , \tag{3a}$$
$$k_{0,t}(x, x') = k_0(x, x') - k_0(x_{1:t}, x)^\top \left(K_{0,t} + \lambda I\right)^{-1} k_0(x_{1:t}, x') \, , \tag{3b}$$
$$\sigma^2_{0,t}(x) = k_{0,t}(x, x) \ , \tag{3c}$$

where $k_0(x_{1:t}, x) = [k_0(x_1, x), k_0(x_2, x), \cdots, k_0(x_t, x)]^\top$, $K_{0,t} = (k_0(x, x'))_{x,x' \in X_t}$ and $y_{0,1:t} = [y_{0,1}, y_{0,2}, \cdots, y_{0,t}]^\top$. Similarly, we can derive $\mu_{i,t}(\cdot), k_{i,t}(\cdot,\cdot), \sigma_{i,t}(\cdot)$, $\forall i \in [N]$ for the constraints. We also introduce the maximum information gain for the objective f as in (Srinivas et al., 2012),

$$\gamma_{0,t} := \max_{X \subset \mathcal{X}; |X| = t} \frac{1}{2} \log \left| I + \lambda^{-1} K_{0,X} \right| , \tag{4}$$

where $K_{0,X} = (k_0(x, x'))_{x,x' \in X}$. Similarly, we can introduce the maximum information gain $\gamma_{i,t}, \forall i \in [N]$ for the constraints.

Lemma 2.4 (Theorem 2, (Chowdhury & Gopalan, 2017a)). *Let Assumptions 2.1 and 2.2 hold. For any $\delta \in (0,1)$, with probability at least $1 - \delta/(N+1)$, the following holds for all $x \in \mathcal{X}$ and $1 \leq t \leq T$, $T \in \mathbb{N}$,*

$$|\mu_{0,t-1}(x) - f(x)| \leq \left(B_0 + \sigma \sqrt{2\left(\gamma_{0,t-1} + 1 + \ln((N+1)/\delta)\right)} \right) \sigma_{0,t-1}(x), \tag{5}$$

where $\mu_{0,t-1}(x), \sigma^2_{0,t-1}(x)$ and $\gamma_{0,t-1}$ are as given in Eq. (3) and Eq. (4), and λ set to be $1 + \frac{2}{T}$. ($\mu_{0,0}$ and $\sigma_{0,0}$ are the prior mean and standard deviation, and $\gamma_{0,0} = 0$.)

We remark that we replace the 'δ' in Theorem 2 of (Chowdhury & Gopalan, 2017a) by $\frac{\delta}{N+1}$, which will be useful to derive the confidence interval with $1-\delta$ probability in Corollary 2.6. We can derive similar confidence intervals for the constraint functions. To facilitate the following algorithm design and discussion, we introduce the lower confidence and upper confidence bound functions,

Definition 2.5. For $i \in \{0\} \cup [N]$, $x \in \mathcal{X}$ and $t \in [T]$,

$$l_{i,t}(x) := \mu_{i,t-1}(x) - \beta_{i,t}\sigma_{i,t-1}(x) \ , \tag{6a}$$
$$u_{i,t}(x) := \mu_{i,t-1}(x) + \beta_{i,t}\sigma_{i,t-1}(x) \ , \tag{6b}$$

where $\beta_{i,t} = B_i + \sigma\sqrt{2\left(\gamma_{i,t-1} + 1 + \ln((N+1)/\delta)\right)}$.

We then have the following corollary,

Corollary 2.6. *Let Assumptions 2.1 and 2.2 hold. With probability at least $1 - \delta$, the following holds for all $x \in \mathcal{X}$ and $1 \leq t \leq T$,*

$$g_i(x) \in [l_{i,t}(x), u_{i,t}(x)] \ , \quad \forall i \in [N] \tag{7}$$
$$\textit{and} \quad f(x) \in [l_{0,t}(x), u_{0,t}(x)]. \tag{8}$$

2.3. Performance Metric

Before we design our algorithm, we introduce the performance metric. We are interested in minimizing the gap of $f(x_t)$ to the optimal value $f^* = \min_{x \in \mathcal{X}, g_i(x) \leq 0, \forall i \in [N]} f(x)$, i.e., the instantaneous regret. Instantaneous regret at step t is defined as,

$$r_t = f(x_t) - f^*, \tag{9}$$

where x_t is the point queried by our algorithm in step t. Since f^* is the constrained optimal value, we may sample some infeasible points with even smaller objective values than f^*, we further define the positive regret as,

$$r_t^+ = [f(x_t) - f^*]^+, \tag{10}$$

where $[\cdot]^+ := \max\{0, \cdot\}$. We are also interested in the constraint violation, defined as,

$$v_{i,t} = [g_i(x_t)]^+. \tag{11}$$

We then introduce cumulative regret that measures the extra cost incurred during the running of the algorithm, as in existing kernelized multi-armed bandit literature (e.g., (Srinivas et al., 2012; Chowdhury & Gopalan, 2017a)).

Definition 2.7 (Cumulative-Regret)**.** We define the cumulative-regret as

$$R_T = \sum_{t=1}^{T} (f(x_t) - f^*). \tag{12}$$

To facilitate the following discussion, we further define the cumulative positive regret,

$$R_T^+ = \sum_{t=1}^{T} [f(x_t) - f^*]^+. \tag{13}$$

We further define the cumulative violation.

Definition 2.8 (Cumulative-Violation)**.** We define the cumulative-violation as

$$\mathcal{V}_{i,T} = \sum_{t=1}^{T} [g_i(x_t)]^+, \forall i \in [N]. \tag{14}$$

Cumulative regret/violations measure the performance during the running of the algorithm. Besides them, we are also interested in the convergence speed to the constrained global optimum. Intuitively, small average regret/violations indicate the existence of a sample with small regret/violation.

3. Algorithm

We propose the lower confidence bound based **CON**strained ef**FI**cient **G**lobal (**CONFIG**) optimization algorithm to solve our problem as shown in Alg. 1 [2]. Our algorithm

Algorithm 1 Lower confidence bounds based **CON**strained ef**FI**cient **G**lobal (**CONFIG**) optimization algorithm.

1: **for** $t \in [T]$ **do**
2: **if** $\exists i \in [N]$, such that $\min_{x \in \mathcal{X}} l_{i,t}(x) > 0$ **then**
3: **Declare infeasibility**.
4: **end if**
5: Find $x_t \in \arg\min_{x \in \mathcal{X}} l_{0,t}(x)$
subject to $l_{i,t}(x) \leq 0, \forall i \in [N]$.
6: Make noisy evaluations of f and $g_i, i \in [N]$ at x_t.
7: Update Gaussian process posterior mean and covariance with the new evaluations added.
8: **end for**

is conceptually simple based on the *optimism in the face of uncertainty* and only requires one to solve an auxiliary problem with the black-box objective and constraints replaced by their lower confidence bound surrogates as shown in line 1 of Alg. 1. Since we consider problems with expensive-to-evaluate functions, the cost of solving the auxiliary problem, which may itself be non-convex, is much smaller compared to solving the original black-box optimization problem when the dimension of the input space is small to medium size. To solve the auxiliary problem, we can use a pure grid search method when the dimension is small (e.g., ≤ 5). For larger dimensions, one can, for example, start from multiple different points and apply gradient-based methods.

Note that in line 1 of Alg. 1, we may declare infeasibility if the auxiliary problem in line 1 is infeasible. When the original problem (1) is feasible, in some very rare cases, we may find the auxiliary problem infeasible due to the potentially unbounded noise. Fortunately, we will show that with guaranteed high probability, line 1 of Alg. 1 is never declared when the original problem is feasible and Assumptions 2.1 and 2.2 hold.

4. Analysis

We now analyze the performance of Alg. 1. All the proofs to the results are put in the appendix. We first introduce a lemma to bound the instantaneous regret and violation,

Lemma 4.1. *Let Assumptions 2.1, 2.2, and 2.3 hold. With probability at least $1 - \delta$, we have first that infeasibility is never declared in line 1 of the Alg. 1, and for all $1 \leq t \leq T$,*

$$r_t \leq r_t^+ \leq 2\beta_{0,t}\sigma_{0,t-1}(x_t) \ , \tag{15}$$

$$v_{i,t} \leq 2\beta_{i,t}\sigma_{i,t-1}(x_t) \ , \quad i \in [N]. \tag{16}$$

We can also bound the cumulative sum of the posterior standard deviation,

Lemma 4.2 (Lemma 4, (Chowdhury & Gopalan, 2017b))**.** *If $x_1, x_2, \cdots, x_T$ are the points selected by Alg. 1, then,*

$$\sum_{t=1}^{T} \sigma_{i,t-1}(x_t) \leq \sqrt{4(T+2)\gamma_{i,T}}. \tag{17}$$

The bound in Eq. (17) is in terms of maximum information gain. To derive a kernel-specific bound, we need to estimate the maximum information gain. A set of upper bounds for the maximum information gains were given by (Srinivas et al., 2012; Vakili et al., 2021), which we restate in the maximum information gain column of Tab. 2. We can now state our main theorem,

Theorem 4.3. *Under Assumptions 2.1, 2.2 and 2.3, we have, with probability at least $1 - \delta$, the sample points of Algorithm 1 satisfy,*

$$R_T \leq R_T^+ \leq 4\beta_{0,T}\sqrt{(T+2)\gamma_{0,T}} = \mathcal{O}(\gamma_{0,T}\sqrt{T}) \ , \tag{18a}$$

$$\mathcal{V}_{i,T} \leq 4\beta_{i,T}\sqrt{(T+2)\gamma_{i,T}} = \mathcal{O}(\gamma_{i,T}\sqrt{T}) \ , \quad \forall i \in [N]. \tag{18b}$$

[2] Knowledge of T is assumed. Standard 'doubling trick' can be applied to extend to the setting without knowing T.

Table 2. Kernel-specific cumulative regret/violation bounds and the convergence rate to the optimal solution. See Appendix A for the specific kernel functions. The column of maximum information gain shows the results of (Srinivas et al., 2012; Vakili et al., 2021). In the table, d represents the dimension of the input space, and ν represents the smoothness parameter of the Mátern kernel.

Kernel	Maximum Information Gain	Cumulative Regret/Violation	Convergence Rate
Linear	$\mathcal{O}(d\log T)$	$\mathcal{O}\left(d\log T\sqrt{T}\right)$	$\mathcal{O}\left(\frac{(N+1)d\log T}{\sqrt{T}}\right)$
Squared Exponential	$\mathcal{O}((\log T)^{d+1})$	$\mathcal{O}\left((\log T)^{d+1}\sqrt{T}\right)$	$\mathcal{O}\left(\frac{(N+1)(\log T)^{d+1}}{\sqrt{T}}\right)$
Mátern $\left(\nu > \frac{d}{2}\right)$	$\mathcal{O}(T^{\frac{d}{2\nu+d}}\log^{\frac{2\nu}{2\nu+d}}(T))$	$\mathcal{O}(T^{\frac{2\nu+3d}{4\nu+2d}}\log^{\frac{2\nu}{2\nu+d}}(T))$	$\mathcal{O}\left((N+1)T^{-\frac{2\nu-d}{4\nu+2d}}\log^{\frac{2\nu}{2\nu+d}}(T)\right)$

Interestingly, the regret bound in Thm. 4.3 is exactly the *same* as the regret bound in the unconstrained case shown in (Chowdhury & Gopalan, 2017b). Furthermore, the constraint violation bound is similar to the regret bound. Based on Thm. 4.3, we can derive a convergence rate to the optimal feasible solution.

Theorem 4.4. *Under Assumptions 2.1, 2.2 and 2.3, we have, with probability at least $1-\delta$, that there exists $\tilde{x}_T \in \{x_1, x_2, \cdots, x_T\}$, such that, $\forall j \in [N]$,*

$$f(\tilde{x}_T) - f^* \leq \frac{4\sqrt{T+2}\sum_{i=0}^{N}\beta_{i,T}\sqrt{\gamma_{i,T}}}{T} = \mathcal{O}\left(\frac{\sum_{i=0}^{N}\gamma_{i,T}}{\sqrt{T}}\right), \tag{19a}$$

$$[g_j(\tilde{x}_T)]^+ \leq \frac{4\sqrt{T+2}\sum_{i=0}^{N}\beta_{i,T}\sqrt{\gamma_{i,T}}}{T} = \mathcal{O}\left(\frac{\sum_{i=0}^{N}\gamma_{i,T}}{\sqrt{T}}\right). \tag{19b}$$

Thm. 4.4 gives the convergence rate to the constrained optimal value in terms of maximum information gain for general kernels. We can now apply Thm. 4.3 and Thm. 4.4 to derive kernel-specific results. Specifically, if $k_i, i \in \{0\} \cup [N]$ share the same kernel types, we can list the kernel-specific bounds and the convergence rates in Tab. 2.

5. Infeasibility Declaration

Thm. 4.3 and Thm. 4.4 are derived under the assumption that the problem is feasible. However, the underlying problem may indeed sometimes be infeasible and therefore, we need an infeasibility detection scheme. In the following, we show that if the original problem is infeasible, we are able to declare infeasibility in line 1 of Alg. 1 and detect the original problem's infeasibility with high probability.

For infeasibility declaration, we try to avoid 'false positives', where a feasible problem is declared as infeasible, and 'false negatives', where an infeasible problem is not declared to be infeasible. Lem. 4.1 already shows that if the original black-box optimization problem is feasible, then with high probability infeasibility is not declared. The following theorem complements Lem. 4.1 by showing that if the original problem is infeasible, our algorithm is guaranteed to declare infeasibility within finite steps with high probability.

Theorem 5.1. *Let Assumptions 2.1 and 2.2 hold. Assume that Prob.* (1) *is infeasible, that is, there exists $i \in [N]$, such that*

$$\min_{x\in\mathcal{X}} \; g_i(x) = \epsilon > 0,$$

and $\lim_{T\to\infty}\frac{\gamma_{i,T}}{\sqrt{T}} = 0$. Given a desired confidence level $\delta \in (0,1)$, Alg. 1 will declare infeasibility on line 1 within a number of steps equal to

$$\overline{T} = \min_{T\in\mathbb{N}_+}\left\{T \,\middle|\, \frac{\gamma_{i,T}}{\sqrt{T}} \leq C\epsilon\right\}, \tag{20}$$

with probability at least $1-\delta$, where C is a constant independent of T.

Remark 3. The proposed algorithm design and analysis in this paper focuses on the case where T is given. In cases where T is not given or not large enough to declare infeasibility, we can apply the 'doubling trick' (Besson & Kaufmann, 2018). The basic idea is that we can start running a round of steps with $T = 1$ and double T after each round. As T grows larger and larger, it will finally be larger than $\overline{T}$ and enough for infeasibility declaration.

Remark 4. Note that in Thm. 5.1, we assume that $\lim_{T\to\infty}\frac{\gamma_{i,T}}{\sqrt{T}} = 0$. From Tab. 2, we can see that for linear and squared exponential kernels, it is satisfied and for Mátern kernel, we require that $\nu > \frac{d}{2}$.

6. Experiments

To demonstrate the effectiveness of CONFIG, we run experiments over sampled instances from the Gaussian process, artificial numerical instances, and a room temperature controller tuning problem. We compare our method to the state-of-the-art Gaussian process optimization baseline methods with constraints consideration. The baselines include SafeOPT (Sui et al., 2015), CEI (**C**onstrained **E**xpected **I**mprovement) based method (Gelbart et al., 2014; Gardner et al., 2014), primal-dual algorithm (Zhou & Ji, 2022) and EPBO (exact penalty Bayesian optimization) (Lu & Paulson, 2022). The experiments are implemented in python, based on the package GPy (GPy, since 2012).

Choice of Parameters. Definition 2.5 gives a rigorous method for selecting the coefficient $\beta_{i,t}$. Since B_i and σ

may be hard to obtain in practice, $\beta_{i,t}$ can usually be set as a constant. The choice of $\beta_{i,t}$ is a key factor that impacts the performance of different algorithms. Empirically, if $\beta_{i,t}$ is set too small, exploration may be insufficient and we may miss the global optimal solution. If $\beta_{i,t}$ is set too large, the algorithm overly explores, converges more slowly, and may suffer from high cumulative regret/violation before converging to the optimal solution. In our experiments shown in this section, manually setting $\beta_{i,t} = 3$ works well. We also set $\lambda = 0.05^2$ as the noise variance for the Gaussian process modelling. We use the common squared exponential kernel functions. For Sec. 6.2 and Sec. 6.3, we randomly sample a few points and maximize the likelihood function to get the hyperparameters of the kernel.

Computational Time. In our experiments, all the problems have low-dimensional input (≤ 3). So we use pure grid search to solve the auxiliary problem for different algorithms and therefore, the computational time is almost the same for different algorithms. With the explicit expression of the lower confidence bounds, the computation cost of solving the auxiliary problems is much cheaper than evaluating the expensive functions, e.g., energy consumption and discomfort of a simulated building in Sec. 6.3.

6.1. Numerical Results over Sampled Instances from Gaussian Process

In this section, we consider the constrained problem with both the objective and the constraint sampled from a Gaussian process. We use the squared exponential kernel as shown in (21).

$$k(x, y) = \sigma_{\mathrm{SE}}^2 \exp\left\{-\frac{\|x-y\|^2}{l^2}\right\}, \tag{21}$$

where $\sigma_{\mathrm{SE}}^2 = 2.0$ and $l = 1.0$. We remove those instances with empty feasible sets. We sampled 48 instances in total.

As shown in Fig. 2, we observe that SafeOPT has almost linear cumulative regret, although achieving close-to-zero violation. This is due to the possibility of getting stuck in a local minimum for SafeOPT. In contrast, the primal-dual method can achieve competitive performance with the state-of-the-art constrained expected improvement (CEI) method in terms of cumulative regret but suffers from almost linear cumulative violation. The EPBO method may also suffer from almost linear cumulative violation if the penalty term is not set large enough. The proposed CONFIG algorithm can achieve performance competitive with the state-of-the-art constrained expected improvement method in terms of both cumulative regret and cumulative violation.

We further show the evolution of the best-observed suboptimality plus violation up to the current step t in Fig. 3, which measures the empirical convergence speed to the op-

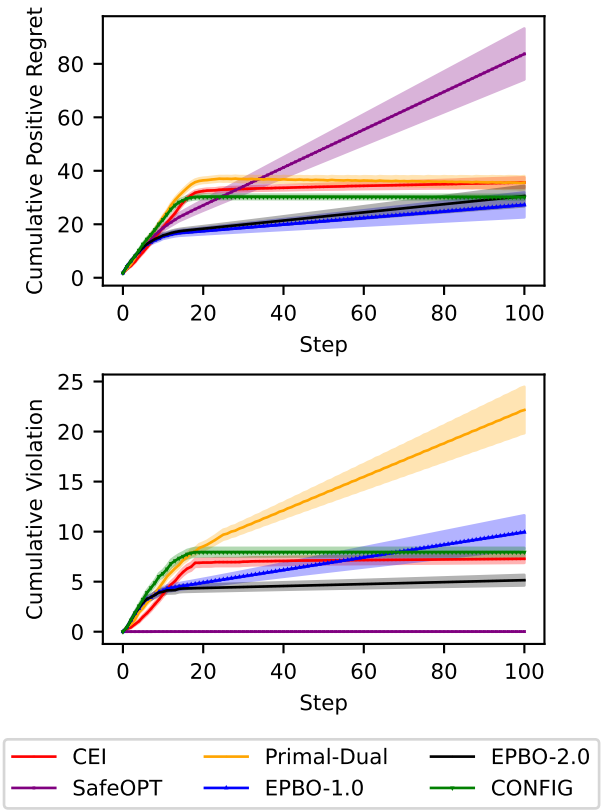

Figure 2. Cumulative regret and violation of different algorithms. The shaded area represents ± 0.1 standard deviation and EPBO-ρ represents EPBO with penalty ρ.

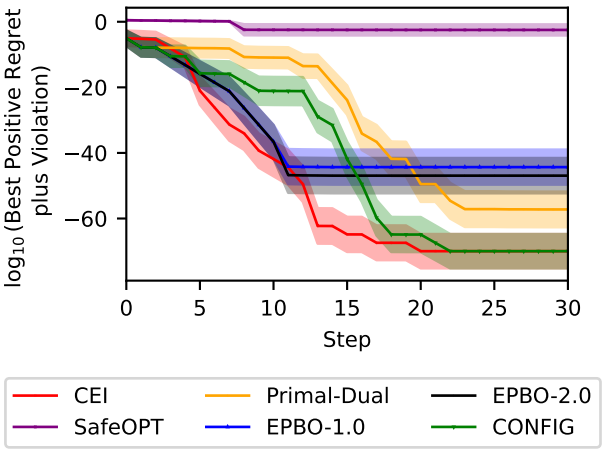

Figure 3. The best suboptimality plus violation up to step t, that is $\min_{\tau \in [t]} [f(x_\tau) - f^*]^+ + [g(x_\tau)]^+$, of different algorithms. The shaded area represents ± 0.1 standard deviation. Here, EPBO-ρ represents the EPBO algorithm with penalty term ρ. The SafeOPT has a higher initial value due to the restriction of starting from a feasible solution.

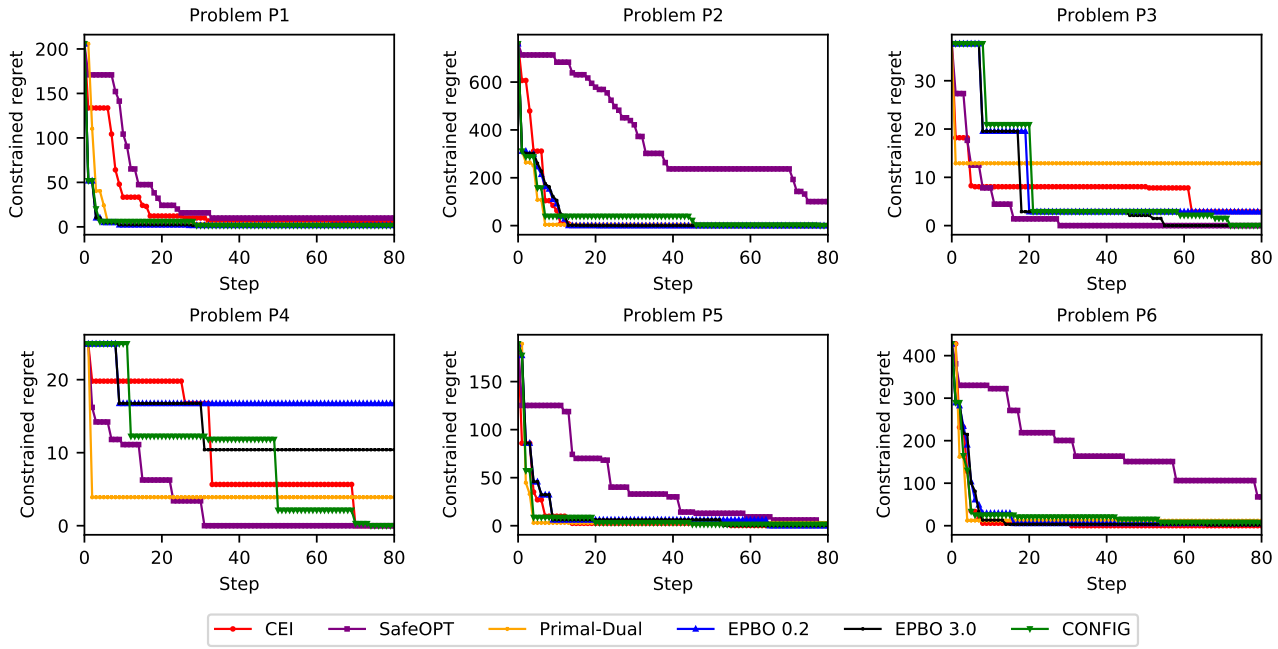

Figure 4. Convergence of constrained regret for the collections of problems in Tab. 3.

timal feasible solution. We notice that our algorithm again achieves empirical convergence speed competitive with the state-of-the-art CEI method.

We then test our algorithms on a set of infeasible instances, where we shift the sampled constraint function $g(x)$ by $\epsilon - \min_{x\in\mathcal{X}} g(x)$ with $\epsilon = 0.1$. Over 50 such infeasible instances, our algorithm consistently declares infeasibility, within 16.3 steps on average.

6.2. Artificial Numerical Instances

In this section, we present the results of a set of global optimization test problems. We consider the global optimization problem in two-dimensional space with $\mathcal{X} = [-10, 10]^2$. We then introduce a set of potentially non-convex and multi-modal functions as in Tab. 4 in the Appendix G. Using these functions as the objective or constraint, we construct a set of global optimization benchmark problems as shown in Tab. 3 (Similar problems in (Vinod et al., 2022)). We use the common squared exponential kernel for all the experiments.

We use constrained regret defined as

$$\min_{\tau\in[t]}[f(x_\tau) - f^*]^+ + [g(x_\tau)]^+, \tag{22}$$

to measure the quality of the solution found by different algorithms up to step t. It can be seen that the smaller the constrained regret is, the closer to the constrained optimum our solution is in terms of optimality and violations.

Fig. 4 demonstrates the convergence of constrained regret

Table 3. Artificial numerical problems constructed using the functions from Tab. 4, where $\mathsf{Qr}_{\mathcal{X}}(h) := \frac{3}{4}\min_{x\in\mathcal{X}} h(x) + \frac{1}{4}\max_{x\in\mathcal{X}} h(x)$.

Problem	Objective f	Constraint g
P1	Br	$\mathrm{SinQ} - \mathsf{Qr}_{\mathcal{X}}(\mathrm{SinQ})$
P2	MBr	$\mathrm{SinQ} - \mathsf{Qr}_{\mathcal{X}}(\mathrm{SinQ})$
P3	Br	$\mathrm{InvBowl} - \mathsf{Qr}_{\mathcal{X}}(\mathrm{InvBowl})$
P4	MBr	$\mathrm{InvBowl} - \mathsf{Qr}_{\mathcal{X}}(\mathrm{InvBowl})$
P5	Br	$\mathrm{Bowl} - \mathsf{Qr}_{\mathcal{X}}(\mathrm{Bowl})$
P6	MBr	$\mathrm{Bowl} - \mathsf{Qr}_{\mathcal{X}}(\mathrm{Bowl})$

for different algorithms. The results show that the CONFIG algorithm consistently achieves a superior or comparable speed of finding the constrained optimal solution, as compared to the other state-of-the-art baselines. We can also notice that in Problem 3 and Problem 4, SafeOPT outperforms both CEI and our method. Intuitively, it happens when the constrained global optimum lies in the same local feasible set as the initial points. In such a case, our method may spend lots of samples exploring globally, while SafeOPT identifies the local (and global) optimal solution quickly by restricting to the local feasible set.

6.3. Tuning the P Controller of a Building

In this section, we consider a building temperature controller tuning problem. We use Energym (Scharnhorst et al., 2021) with a lower level Modelica (Fritzson & Engelson, 1998)

model with a single thermal zone as our simulator. We consider a P controller controlling the temperature. The mathematical expression of the P controller is given as

$$u = K_p(\text{set-point} - \text{temperature}),$$

where K_p is the proportional gain and u is the heating control. We use θ to denote the tuning parameters, which include the proportional gain, daytime set-point for temperature, and the switching time from nighttime set-point to daytime set-point. Here, the nighttime set-point is fixed. Our objective is to minimize energy consumption while keeping the average temperature deviation below a given threshold. Our problem is formulated as in Eq. (23).

$$\min_{\theta \in \Theta} \quad J(\theta) \quad \text{subject to} \quad g(\theta) - g_{\text{thr}} \leq 0, \tag{23}$$

where $J(\theta)$ represents energy consumption in this example, $g(\theta)$ represents average temperature deviation and g_{thr} is a given average temperature deviation threshold. In each step, our algorithm selects a set of promising controller parameters. We use this set of parameters to run a closed-loop building simulation over a period of 1 day.

Based on the closed-loop trajectory, we calculate the key performance indicators, that is, energy consumption and average temperature deviation. We then update the Gaussian process regression with the new data added. Since energy consumption and temperature deviations are in two different scales, we use normalized energy consumption $\frac{J}{\sigma_J}$ plus normalized average temperature deviation $\frac{[g-g_{\text{thr}}]^+}{\sigma_g}$ to measure the quality of the solution, where σ_J, σ_g are standard deviations in a set of randomly sampled data.

Fig. 5 gives the best-normalized energy consumption plus average temperature deviation up to the current step. We observe that within 15 steps, our algorithm identifies the solution with minimized normalized energy consumption plus normalized average temperature deviation, verifying the competitiveness of our algorithm with the currently popular constrained expected improvement method again. In contrast, the SafeOPT method can be overly cautious and fails to find a solution with performance comparable to ours.

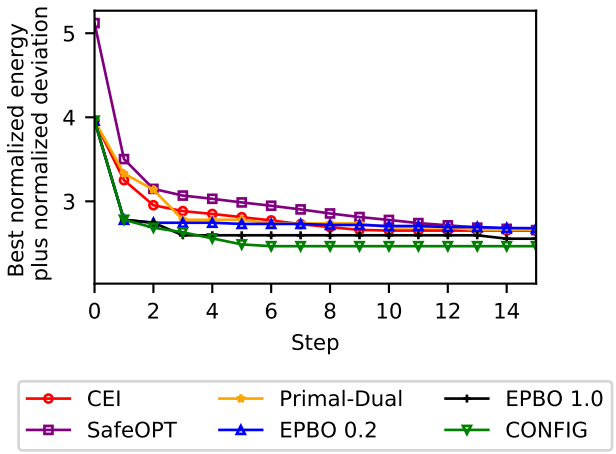

Figure 5. Best normalized energy consumption plus normalized temperature deviation up to the current step. For SafeOPT, the initial required feasible solution is obtained based on domain knowledge.

7. Conclusion

In this paper, we have presented CONFIG, a simple and effective algorithm, for constrained efficient global optimization of expensive black-box functions by exploiting the principle of *optimism in the face of uncertainty*. Specifically, our algorithm solves an auxiliary constrained optimization problem with the lower confidence bound (LCB) surrogates as objective and constraints to get the next sample. Theoretically, we show that our algorithm enjoys the same cumulative regret bound as that in the unconstrained case. In addition, we show that the cumulative constraint violations have upper bounds in terms of maximum information gain, which are similar to the cumulative regret bounds of the objective function. For commonly used Mátern ($\nu > \frac{d}{2}$) and Squared Exponential kernels, our bounds are sublinear and allow us to derive a convergence rate to the optimal solution of the original constrained problem. Furthermore, our method naturally provides a scheme to declare infeasibility when the original black-box optimization problem is infeasible. Numerical experiments over sampled instances, artificial problems, and a building controller tuning problem, all corroborate the effectiveness of our algorithm, which is competitive with the popular CEI method. As compared to existing state-of-the-art constrained efficient global/Bayesian optimization methods, CONFIG significantly improves the theoretical guarantees, while achieving competitive empirical performance.

One limitation of our work is that the auxiliary problem may be infeasible with misspecified hyperparameters, which needs to be addressed in future work.

Acknowledgements

This research was supported by the Swiss National Science Foundation under NCCR Automation, grant agreement 51NF40_180545, and in part by the Swiss Data Science Center, grant agreement C20-13.

References

Amani, S., Alizadeh, M., and Thrampoulidis, C. Linear stochastic bandits under safety constraints. *Advances in Neural Information Processing Systems*, 32, 2019.

Amani, S., Alizadeh, M., and Thrampoulidis, C. Regret bound for safe Gaussian process bandit optimization. In *Learning for Dynamics and Control*, pp. 158–159. PMLR, 2020.

Audibert, J.-Y., Bubeck, S., and Munos, R. Best arm identification in multi-armed bandits. In *The Annual Conference On Learning Theory (COLT)*, pp. 41–53. Citeseer, 2010.

Bagheri, S., Konen, W., Allmendinger, R., Branke, J., Deb, K., Fieldsend, J., Quagliarella, D., and Sindhya, K. Constraint handling in efficient global optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 673–680, 2017.

Bansal, S., Calandra, R., Xiao, T., Levine, S., and Tomlin, C. J. Goal-driven dynamics learning via Bayesian optimization. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 5168–5173. IEEE, 2017.

Basudhar, A., Dribusch, C., Lacaze, S., and Missoum, S. Constrained efficient global optimization with support vector machines. *Structural and Multidisciplinary Optimization*, 46(2):201–221, 2012.

Bergstra, J. and Bengio, Y. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(2), 2012.

Besson, L. and Kaufmann, E. What doubling tricks can and can't do for multi-armed bandits. *arXiv preprint arXiv:1803.06971*, 2018.

Chowdhury, S. R. and Gopalan, A. On kernelized multi-armed bandits. In *International Conference on Machine Learning*, pp. 844–853. PMLR, 2017a.

Chowdhury, S. R. and Gopalan, A. On kernelized multi-armed bandits. *arXiv preprint arXiv:1704.00445*, 2017b.

Frazier, P. I. A tutorial on Bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

Frazier, P. I. and Wang, J. Bayesian optimization for materials design. In *Information Science for Materials Discovery and Design*, pp. 45–75. Springer, 2016.

Fritzson, P. and Engelson, V. Modelica—a unified object-oriented language for system modeling and simulation. In *European Conference on Object-Oriented Programming*, pp. 67–90. Springer, 1998.

Gardner, J. R., Kusner, M. J., Xu, Z. E., Weinberger, K. Q., and Cunningham, J. P. Bayesian optimization with inequality constraints. In *International Conference on Machine Learning (ICML)*, volume 2014, pp. 937–945, 2014.

Gelbart, M. A., Snoek, J., and Adams, R. P. Bayesian optimization with unknown constraints. In *Proc. of the 30th Conf. on Uncertainty in Artif. Intell.*, UAI'14, pp. 250–259, Arlington, Virginia, USA, 2014. AUAI Press. ISBN 9780974903910.

GPy. GPy: A Gaussian process framework in python. `http://github.com/SheffieldML/GPy`, since 2012.

Guo, H., Zhu, Q., and Liu, X. Rectified pessimistic-optimistic learning for stochastic continuum-armed bandit with constraints. *arXiv preprint arXiv:2211.14720*, 2022.

Hernández-Lobato, J. M., Gelbart, M. A., Adams, R. P., Hoffman, M. W., and Ghahramani, Z. A general framework for constrained Bayesian optimization using information-based search. 2016.

Inatsu, Y., Takeno, S., Karasuyama, M., and Takeuchi, I. Bayesian optimization for distributionally robust chance-constrained problem. In *International Conference on Machine Learning*, 2022.

Jeong, S., Murayama, M., and Yamamoto, K. Efficient optimization design method using kriging model. *Journal of Aircraft*, 42(2):413–420, 2005.

Jones, D. R., Schonlau, M., and Welch, W. J. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

Lu, C. and Paulson, J. A. No-regret Bayesian optimization with unknown equality and inequality constraints using exact penalty functions. *IFAC-PapersOnLine*, 55(7):895–902, 2022.

Moradipari, A., Thrampoulidis, C., and Alizadeh, M. Stage-wise conservative linear bandits. *Advances in Neural Information Processing Systems*, 33:11191–11201, 2020.

Negoescu, D. M., Frazier, P. I., and Powell, W. B. The knowledge-gradient algorithm for sequencing experiments in drug discovery. *INFORMS Journal on Computing*, 23(3):346–363, 2011.

Pardalos, P. M. and Romeijn, H. E. *Handbook of Global Optimization: Volume 2*, volume 62. Springer Science & Business Media, 2013.

Sasena, M. J., Papalambros, P., and Goovaerts, P. Exploration of metamodeling sampling criteria for constrained global optimization. *Engineering optimization*, 34(3): 263–278, 2002.

Scharnhorst, P., Schubnel, B., Fernández Bandera, C., Salom, J., Taddeo, P., Boegli, M., Gorecki, T., Stauffer, Y., Peppas, A., and Politi, C. Energym: A building model library for controller benchmarking. *Appl. Sciences*, 11 (8):3518, 2021.

Schölkopf, B., Herbrich, R., and Smola, A. J. A generalized representer theorem. In *International Conference on Computational Learning Theory*, pp. 416–426. Springer, 2001.

Schonlau, M. Computer experiments and global optimization. 1997.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and De Freitas, N. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.

Snoek, J., Rippel, O., Swersky, K., Kiros, R., Satish, N., Sundaram, N., Patwary, M., Prabhat, M., and Adams, R. Scalable Bayesian optimization using deep neural networks. In *International Conference on Machine Learning*, pp. 2171–2180. PMLR, 2015.

Soare, M., Lazaric, A., and Munos, R. Best-arm identification in linear bandits. *Advances in Neural Information Processing Systems*, 27, 2014.

Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. W. Information-theoretic regret bounds for Gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012.

Sui, Y., Gotovos, A., Burdick, J., and Krause, A. Safe exploration for optimization with Gaussian processes. In *Proc. of the Int. Conf. on Mach. Learn.*, pp. 997–1005, 2015.

Sui, Y., Burdick, J., Yue, Y., et al. Stage-wise safe Bayesian optimization with Gaussian processes. In *Proc. of the Int. Conf. on Mach. Learn.*, pp. 4781–4789, 2018.

Turchetta, M., Berkenkamp, F., and Krause, A. Safe exploration for interactive machine learning. *Advances in Neural Inf. Process. Syst. 32*, 4:2868–2878, 2020.

Vakili, S., Khezeli, K., and Picheny, V. On information gain and regret bounds in Gaussian process bandits. In *International Conference on Artificial Intelligence and Statistics*, pp. 82–90. PMLR, 2021.

Vinod, A. P., Israel, A., and Topcu, U. Constrained, global optimization of unknown functions with lipschitz continuous gradients. *SIAM Journal on Optimization*, 32(2): 1239–1264, 2022.

Wu, Y., Shariff, R., Lattimore, T., and Szepesvári, C. Conservative bandits. In *International Conference on Machine Learning*, pp. 1254–1262. PMLR, 2016.

Xu, W., Jiang, Y., Maddalena, E. T., and Jones, C. N. Lower bounds on the worst-case complexity of efficient global optimization. *arXiv preprint arXiv:2209.09655*, 2022a.

Xu, W., Jones, C. N., Svetozarevic, B., Laughman, C. R., and Chakrabarty, A. VABO: Violation-aware Bayesian optimization for closed-loop control performance optimization with unmodeled constraints. In *2022 American Control Conference (ACC)*, pp. 5288–5293. IEEE, 2022b.

Xu, W., Jiang, Y., Svetozarevic, B., and Jones, C. N. Primal-dual contextual Bayesian optimization for control system online optimization with time-average constraints. *arXiv preprint arXiv:2304.06104*, 2023.

Zhang, Y., Zhang, X., and Frazier, P. I. Two-step lookahead Bayesian optimization with inequality constraints. *Advances in Neural Information Processing Systems*, 34, 2021.

Zhou, X. and Ji, B. On kernelized multi-armed bandits with constraints. *Advances in Neural Information Processing Systems*, 35, 2022.

A. Examples of kernel functions

- *Linear*:

$$k(x, y) = x^T y.$$

- *Squared Exponential*:

$$k(x, y) = \exp\left\{-\frac{\|x - y\|^2}{\ell^2}\right\}.$$

- *Mátern*:

$$k(x, y) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\frac{\|x - y\|}{\rho}\right)^{\nu} K_\nu \left(\sqrt{2\nu}\frac{\|x - y\|}{\rho}\right),$$

where ρ and ν are positive parameters of the kernel function, Γ is the gamma function, and K_ν is the modified Bessel function of the second kind.

B. Proof of Corollary 2.6

Proof. Note that $l_{i,t}(x)$ and $u_{i,t}(x)$ are random variables since they depend on the observations $y_{i,t}$'s, which are corrupted by random noise. The following events definitions are with respect to the randomness from noise. For simplicity of notation, we use $\{l_{i,t}(x) \leq y \leq u_{i,t}(x)\}$ to denote the intersection event that the random variable $l_{i,t}(x) \leq y$ and $u_{i,t}(x) \geq y$, where $y \in \mathbb{R}$, $i \in \{0\} \cup [N]$ and $t \geq 1$. We define the event $\mathcal{E}_i \triangleq \cap_{x\in\mathcal{X}} \cap_{T\geq t\geq 1} \{l_{i,t}(x) \leq g_i(x) \leq u_{i,t}(x)\}, \forall i \in [N]$ and the event $\mathcal{E}_0 \triangleq \cap_{x\in\mathcal{X}} \cap_{T\geq t\geq 1} \{l_{0,t}(x) \leq f(x) \leq u_{0,t}(x)\}$. We have,

$$\begin{aligned}
\mathbb{P}\left(\cap_{i=0}^{N}\mathcal{E}_i\right) &= 1 - \mathbb{P}\left(\overline{\cap_{i=0}^{N}\mathcal{E}_i}\right) && \text{(24a)}\\
&= 1 - \mathbb{P}\left(\cup_{i=0}^{N}\overline{\mathcal{E}_i}\right) && \text{(24b)}\\
&\geq 1 - \sum_{i=0}^{N}\mathbb{P}\left(\overline{\mathcal{E}_i}\right) && \text{(24c)}\\
&\geq 1 - \sum_{i=0}^{N}\frac{\delta}{N+1} && \text{(24d)}\\
&= 1 - \delta, && \text{(24e)}
\end{aligned}$$

where the inequality (24c) follows by union bound of probability and the inequality (24d) follows by Lemma 2.4. The conclusion then follows.

□

C. Proof of Lemma 4.1

Proof. By Corollary 2.6, with probability at least $1-\delta$, we have, for all $x \in \mathcal{X}$ and $1 \leq t \leq T$,

$$\begin{aligned}
f(x) &\in [l_{0,t}(x), u_{0,t}(x)] \ , && \text{(25a)}\\
g_i(x) &\in [l_{i,t}(x), u_{i,t}(x)] \ , \quad \forall i \in [N]. && \text{(25b)}
\end{aligned}$$

All the following statements are conditioned on the above joint event happening, which has a probability of at least $1-\delta$.

By setting $x = x^*$ in (25b) and the feasibility of the optimal solution x^*, we have

$$l_{i,t}(x^*) \leq g_i(x^*) \leq 0, \quad \forall i \in [N]. \tag{26}$$

Therefore, infeasibility is not declared in line 1 of the Alg. 1, and x^* is a feasible solution for the auxiliary problem in line 1 of the Alg. 1. Therefore, we have

$$l_{0,t}(x_t) \leq l_{0,t}(x^*). \tag{27}$$

Therefore,

$$\begin{aligned}
r_t &= f(x_t) - f^* && \text{(28a)}\\
&\leq r_t^+ && \text{(28b)}\\
&= [f(x_t) - l_{0,t}(x_t) + l_{0,t}(x_t) - f^*]^+ && \text{(28c)}\\
&\leq [f(x_t) - l_{0,t}(x_t)]^+ + [l_{0,t}(x_t) - f^*]^+ && \text{(28d)}\\
&\leq [u_{0,t}(x_t) - l_{0,t}(x_t)]^+ + [l_{0,t}(x_t) - l_{0,t}(x^*)]^+ && \text{(28e)}\\
&= u_{0,t}(x_t) - l_{0,t}(x_t) && \text{(28f)}\\
&= 2\beta_{0,t}\sigma_{0,t-1}(x_t), && \text{(28g)}
\end{aligned}$$

where the inequality (28e) follows by the inequalities (25a), the inequality (28d) follows by the fact that $[a+b]^+ \leq [a]^+ + [b]^+, \forall a, b \in \mathbb{R}$, and the equality (28f) follows by the inequality (27). We now consider the constraint violation,

$$\begin{aligned}
v_{i,t} &= [g_i(x_t)]^+ && \text{(29a)}\\
&= [g_i(x_t) - l_{i,t}(x_t) + l_{i,t}(x_t)]^+ && \text{(29b)}\\
&\leq [g_i(x_t) - l_{i,t}(x_t)]^+ + [l_{i,t}(x_t)]^+ && \text{(29c)}\\
&= [g_i(x_t) - l_{i,t}(x_t)]^+ && \text{(29d)}\\
&\leq [u_{i,t}(x_t) - l_{i,t}(x_t)]^+ && \text{(29e)}\\
&= 2\beta_{i,t}\sigma_{i,t-1}(x_t), && \text{(29f)}
\end{aligned}$$

where the equality (29d) follows by the feasibility of x_t for the auxilliary problem in line 1 of Alg. 1. □

D. Proof of Theorem 4.3

Proof. By the monotonicity of $\beta_{i,t}$ and Lem. 17, we have,

$$\begin{aligned}
R_T &= \sum_{t=1}^{T} r_t && \text{(30a)}\\
&\leq \sum_{t=1}^{T} r_t^+ = R_T^+ && \text{(30b)}\\
&\leq \sum_{t=1}^{T} 2\beta_{0,t}\sigma_{0,t}(x_t) && \text{(30c)}\\
&\leq 2\beta_{0,T}\sum_{t=1}^{T}\sigma_{0,t}(x_t) && \text{(30d)}\\
&\leq 4\beta_{0,T}\sqrt{(T+2)\gamma_{0,T}}, && \text{(30e)}
\end{aligned}$$

and $\forall i \in [N]$,

$$\mathcal{V}_{i,T} = \sum_{t=1}^{T} v_{i,t} \tag{31a}$$

$$\leq \sum_{t=1}^{T} 2\beta_{i,t}\sigma_{i,t}(x_t) \tag{31b}$$

$$\leq 2\beta_{i,T}\sum_{t=1}^{T}\sigma_{i,t}(x_t) \tag{31c}$$

$$\leq 4\beta_{i,T}\sqrt{(T+2)\gamma_{i,T}}, \tag{31d}$$

Combining with the expression of $\beta_{i,t}$ in Definition 2.5, we can derive

$$R_T \leq R_T^+ \leq 4\beta_{0,T}\sqrt{(T+2)\gamma_{0,T}} = \mathcal{O}(\gamma_{0,T}\sqrt{T}) \ ,$$

$$\mathcal{V}_{i,T} \leq 4\beta_{i,T}\sqrt{(T+2)\gamma_{i,T}} = \mathcal{O}(\gamma_{i,T}\sqrt{T}) \ , \quad \forall i \in [N],$$

which completes the proof. □

E. Proof of Theorem 4.4

Proof. Consider the sum,

$$\begin{aligned}
&\sum_{t=1}^{T}\left([f(x_t)-f^*]^+ + \sum_{i=1}^{N}[g_i(x_t)]^+\right)\\
=&R_T^+ + \sum_{i=1}^{N}\mathcal{V}_{i,T}\\
\leq&4\sqrt{T+2}\sum_{i=0}^{N}\beta_{i,T}\sqrt{\gamma_{i,T}},
\end{aligned}$$

where the last inequality follows by the inequalities (30e) and (31d). Therefore,

$$\begin{aligned}
&\min_{t\in[T]}\left([f(x_t)-f^*]^+ + \sum_{i=1}^{N}[g_i(x_t)]^+\right)\\
\leq&\frac{1}{T}\sum_{t=1}^{T}\left([f(x_t)-f^*]^+ + \sum_{i=1}^{N}[g_i(x_t)]^+\right)\\
\leq&\frac{4\sqrt{T+2}\sum_{i=0}^{N}\beta_{i,T}\sqrt{\gamma_{i,T}}}{T}.
\end{aligned}$$

So there exists $\tilde{x}_T \in \{x_1, x_2, \cdots, x_T\}$, such that,

$$[f(\tilde{x}_T)-f^*]^+ + \sum_{i=1}^{N}[g_i(\tilde{x}_T)]^+ \leq \frac{4\sqrt{T+2}\sum_{i=0}^{N}\beta_{i,T}\sqrt{\gamma_{i,T}}}{T}.$$

Since $[f(\tilde{x}_T)-f^*]^+$ and $[g_i(\tilde{x}_T)]^+$ are non-negative, we have

$$f(\tilde{x}_T)-f^* \leq [f(\tilde{x}_T)-f^*]^+ \leq \frac{4\sqrt{T+2}\sum_{i=0}^{N}\beta_{i,T}\sqrt{\gamma_{i,T}}}{T}$$

$$[g_j(\tilde{x}_T)]^+ \leq \frac{4\sqrt{T+2}\sum_{i=0}^{N}\beta_{i,T}\sqrt{\gamma_{i,T}}}{T} \ , \quad \forall j \in [N].$$

The desired conclusion then follows. □

F. Proof of Theorem 5.1

Proof. Suppose up to step T, infeasibility has not been declared. We have,

$$l_{i,t}(x_t) \leq 0$$

and

$$u_{i,t}(x_t) \geq g_i(x_t) \geq \epsilon$$

with probability at least $1-\delta$.

Accordingly, we have

$$2\sum_{t=1}^{T}\beta_{i,t}\sigma_{i,t-1}(x_t) = \sum_{t=1}^{T}(u_{i,t}(x_t)-l_{i,t}(x_t)) \geq \sum_{t=1}^{T}\epsilon = T\epsilon.$$

Meanwhile,

$$\begin{aligned}
2\sum_{t=1}^{T}\beta_{i,t}\sigma_{i,t-1}(x_t) &\leq 2\beta_{i,T}\sum_{t=1}^{T}\sigma_{i,t-1}(x_t)\\
&\leq 2\beta_{i,T}\sqrt{4(T+2)\gamma_{i,T}},
\end{aligned}$$

where the last inequality follows by Lem. 17. Therefore,

$$2\beta_{i,T}\sqrt{4(T+2)\gamma_{i,T}} \geq T\epsilon,$$

which implies $\epsilon \leq \frac{2\beta_{i,T}\sqrt{4(T+2)\gamma_{i,T}}}{T}$. By Def. 2.5, we have

$$\frac{2\beta_{i,T}\sqrt{4(T+2)\gamma_{i,T}}}{T} = \mathcal{O}\left(\frac{\gamma_{i,T}}{\sqrt{T}}\right).$$

Hence, $\exists \tilde{C} > 0$, such that,

$$\epsilon \leq \tilde{C}\frac{\gamma_{i,T}}{\sqrt{T}}. \tag{32}$$

That is

$$C\epsilon \leq \frac{\gamma_{i,T}}{\sqrt{T}}, \tag{33}$$

where $C = \frac{1}{\tilde{C}}$. However, since $\lim_{T\to\infty}\frac{\gamma_{i,T}}{\sqrt{T}} = 0$, the inequality (33) will be violated when T is large enough. So infeasibility will be declared on or before the first time the above inequality is violated, which is $\overline{T}$. □

G. Explicit functions for the artificial numerical instances

Name	Function
Branin (Br)	$\mathrm{Br}\left(x_1, x_2\right) = \left(x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos\left(x_1\right) + 10\ .$
Modified Branin (MBr)	$\mathrm{MBr}\left(x_1, x_2\right) = \mathrm{Br}\left(x_1, x_2\right) + 20x_1 - 30x_2\ .$
Bowl (Bowl)	$\mathrm{Bowl}\ \left(x_1, x_2\right) = \frac{1}{2}\left(\left\|x - c_{\text{bowl}}\right\|^2 - R_{\text{bowl}}^2\right)$ where $R_{\text{bowl}} = 10$ and $c_{\text{bowl}} = [-3, -3]$.
Inverted bowl (InvBowl)	$\mathrm{InvBowl}\left(x_1, x_2\right) = -\mathrm{Bowl}\left(x_1, x_2\right)\ .$
Sine-quadratic (SinQ)	$\mathrm{SinQ}\left(x_1, x_2\right) = \sin\left(\frac{x_1^2 + x_2^2}{10}\right)\ .$

Table 4. Explicit functions used to construct the artificial numerical instances.