Scalable Constrained Bayesian Optimization

David Eriksson[1]
Facebook
deriksson@fb.com

Matthias Poloczek[1]
Amazon
matpol@amazon.com

Abstract

The global optimization of a high-dimensional black-box function under black-box constraints is a pervasive task in machine learning, control, and engineering. These problems are challenging since the feasible set is typically non-convex and hard to find, in addition to the curses of dimensionality and the heterogeneity of the underlying functions. In particular, these characteristics dramatically impact the performance of Bayesian optimization methods, that otherwise have become the de facto standard for sample-efficient optimization in unconstrained settings, leaving practitioners with evolutionary strategies or heuristics. We propose the scalable constrained Bayesian optimization (`SCBO`) algorithm that overcomes the above challenges and pushes the applicability of Bayesian optimization far beyond the state-of-the-art. A comprehensive experimental evaluation demonstrates that `SCBO` achieves excellent results on a variety of benchmarks. To this end, we propose two new control problems that we expect to be of independent value for the scientific community.

1 Introduction

The global optimization of a black-box objective function under black-box constraints has many applications in machine learning, engineering, and the natural sciences. Examples include fine-tuning the efficiency of a computing platform while preserving the quality of service; maximizing the power conversion efficiency of a solar cell material under stability and reliability requirements; optimizing the control policy of a robot under performance and safety constraints; tuning the performance of an aerospace design averaged over multiple scenarios while ensuring a satisfactory performance on each individual scenario (multi-point optimization). Moreover, a popular approach for multi-objective optimization tasks is to to reformulate them as constrained problems. Here the functions that comprise the objective and the constraints are often given as black-boxes, i.e., upon their evaluation we receive an observation of the respective function, possibly with noise but without derivative information. All of the above examples have in common that their dimensionality, that is, the number of tunable parameters, is large: it is usually up to several dozens, which poses a substantial challenge for current methods in derivative-free optimization.

High dimensionality makes black-box functions hard to optimize due to the curses of dimensionality [Powell, 2019], even in the absence of constraints. Moreover, these functions are often heterogeneous which poses a problem for surrogate-based optimizers. Black-box constraints make the task considerably harder since the set of feasible points is typically non-convex and hard to find, e.g., for control problems.

The main contributions of this work are as follows:

1. We propose the *scalable constrained Bayesian optimization* algorithm (`SCBO`), the first scalable algorithm for the optimization of high-dimensional expensive functions under expensive constraints. `SCBO` is also the first algorithm to support large batches for constrained problems with native support for asynchronous observations.

2. A comprehensive evaluation shows that `SCBO` outperforms previous state-of-the-art methods by far on high-dimensional constrained problems. Moreover, `SCBO` at least matches and often beats the best performer on low-dimensional instances.

3. We introduce two new high-dimensional con-

Proceedings of the 24$^{\text{th}}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

1. This work was conducted while the authors were affiliated with Uber AI.

strained test problems that will be of independent interest given the novelty and anticipated impact of large-scale constrained Bayesian optimization.

1.1 Related Work

Bayesian optimization (BO) has recently gained enormous popularity for the global optimization of expensive black-box functions, see [Frazier, 2018, Shahriari et al., 2016] for an overview. While the vast majority of work focuses on unconstrained problems, aside from box constraints that describe the search space, a handful of articles consider the presence of black-box constraints. The seminal work of Schonlau et al. [1998] extends the expected improvement criterion (`EI`) to constraints by multiplying the expected improvement at some point x over the best *feasible* point with the probability that x itself is feasible, leveraging the independence between the objective function and the constraints. Later this `cEI` algorithm was rediscovered by [Gardner et al., 2014, Gelbart et al., 2014] and studied in a variety of settings, e.g., see [Sóbester et al., 2014, Forrester et al., 2008, Parr et al., 2012b,a] and the references therein. Letham et al. [2019] extended the approach to noisy observations using quasi Monte Carlo integration and were the first to consider batch acquisition under constraints. Note that for noise-free observations, as for the benchmarks that we study, their approach reduces to the original `cEI`. Bayesian optimization with constraints was also studied in the context of lookahead acquisition and with multiple information sources [Lam et al., 2015, Lam and Willcox, 2017, Lam et al., 2018].

Hernández-Lobato et al. [2016] extended predictive entropy search [Hernández-Lobato et al., 2014] to constraints and detailed how to make the sophisticated approximation of the entropy reduction computationally tractable in practice. Their `PESC` algorithm usually achieves great results and is widely considered the state-of-the-art for constrained BO despite its rather large computational costs. Picheny [2014] considered the volume of the admissible excursion set under the best known feasible point as a measure for the uncertainty over the location of the optimizer. His algorithm iteratively samples a point that yields a maximum approximate reduction in volume.

By lifting constraints into the objective via the Lagrangian relaxation, Gramacy et al. [2016] took a different approach. Note that it results in a series of unconstrained optimization problems that are solved by vanilla BO. `SLACK` of Picheny et al. [2016] refined this idea by introducing slack variables and showed that this augmented Lagrangian achieves a better performance for *equality* constraints. Very recently, Ariafar et al. [2019] used the `ADMM` algorithm to solve an augmented Lagrangian relaxation. All these algorithms use the `EI` criterion.

Traditionally, BO, with or without constraints, has been limited to problems with a small number of decision variables, usually at most 15, and a budget of no more than a couple of hundred samples. Recent work has started exploring scalable BO for budgets with tens of thousands of samples. Hernández-Lobato et al. [2017] extended Thompson sampling [Thompson, 1933] to large batch sizes and used a Bayesian neural network for the surrogate to maintain scalability (see also [Kandasamy et al., 2018]). Wang et al. [2018] proposed the `EBO` algorithm that partitions the search space to achieve scalability. Eriksson et al. [2019] abandoned a global surrogate and instead maintained several local models that move towards better solutions. Their `TURBO` algorithm applies a bandit approach to allocate samples efficiently between these local searches. Independently, Mathesen et al. [2020] also proposed to combine trust region modeling with Bayesian optimization with an `EI`-based acquisition criterion to balance global and local optimization. BO has been investigated for high-dimensional settings with small sampling budgets, e.g., see [Wang et al., 2016, Binois et al., 2015, Eriksson et al., 2018, Mutny and Krause, 2018, Oh et al., 2018, Rolland et al., 2018, Kirschner et al., 2019, Nayebi et al., 2019, Letham et al., 2020, Binois et al., 2020]. Bouhlel et al. [2018] combined a dimensionality reduction via partial least squares with kriging-based `EI` to solve a 50-dimensional reduced version of the constrained MOPTA problem. The authors pointed out that their approach cannot handle the full MOPTA problem studied in Sect. 4.

The constrained optimization of black-box functions has also been studied in the field of evolutionary strategies and in operations research. `CMA-ES` is one of the most powerful and versatile evolutionary strategies. It uses a covariance adaptation strategy to learn a second-order model of the objective function. `CMA-ES` handles constraints by the 'death penalty' that sets the fitness value of infeasible solutions to zero [Kramer, 2010, Arnold and Hansen, 2012]. `COBYLA` [Powell, 1994] and `BOBYQA` [Powell, 2007] maintain a local trust region and thus perform a local search. In our experience, this strategy scales well to high dimensions, with `COBYLA` having an edge due to its support for non-linear constraints. We will compare to `cEI` that we extended to high-dimensional domains, `PESC`, `SLACK`, `CMA-ES`, and `COBYLA` and thus have a representative selection of above lines of work. Data-dependent transformations of the black-box functions were studied in [Snelson et al., 2004, Wilson and Ghahramani, 2010, Snoek et al., 2014, Salinas et al., 2019].

Structure of the article. The remainder of the article is structured as follows. In the next section we define the problem formally. The `SCBO` algorithm is presented in Sect. 3 and compared to a representative selection of methods in Sect. 4. Sect. 5 summarizes the conclusions and discusses ideas for future work.

2 The Model

The goal is to find an optimizer

$$\operatorname*{argmin}_{x \in \Omega} f(x) \;\text{ s.t. }\; c_1(x) \leq 0, \ldots, c_m(x) \leq 0 \qquad (1)$$

where $f\colon \Omega \to \mathbb{R}$ and $c_\ell\colon \Omega \to \mathbb{R}$ for $1 \leq \ell \leq m$ are black-box functions defined over a compact set $\Omega \subset \mathbb{R}^d$. The term black-box function means that we may query any $x \in \Omega$ to observe the values under the objective function f and all constraints, possibly with noise, but no derivative information. Specifically, we suppose that we observe an i.i.d. $(m+1)$-dimensional vector with the ℓ-th entry given by $y_0(x) \sim \mathcal{N}(f(x), \lambda_0(x))$ and $y_\ell(x) \sim \mathcal{N}(c_\ell(x), \lambda_\ell(x))$ for $1 \leq \ell \leq m$. Here the λ's give the variance of the observational noise and are supposed to be known. In practice, we estimate the λ's along with the hyperparameters of the surrogate model. Note that we may rescale the search space Ω w.l.o.g. to the unit hypercube $[0, 1]^d$. If all functions are observed without noise, then our goal is to find a feasible point with minimum value under the objective function. For noisy functions, we wish to find a point with best expected objective value under all points that are feasible with probability at least $1-\delta$, where δ is set based on the context, e.g., the degree of risk aversion (cp. Hernández-Lobato et al. [2016]).

3 Scalable Constrained Bayesian Optimization (`SCBO`)

We propose the *Scalable Constrained Bayesian Optimization* (`SCBO`) algorithm. `SCBO` follows the paradigm of the generic BO algorithm [Frazier, 2018, Shahriari et al., 2016] and proceeds in rounds. In each round, `SCBO` selects a batch of q points in Ω that are then evaluated in parallel. Note that `SCBO` is easily extended to asynchronous batch evaluations.

`SCBO` employs the *trust region* approach introduced by Eriksson et al. [2019] that confines samples locally. This addresses common problems of Bayesian optimization in high-dimensional settings, where popular acquisition functions spread out samples due to the inherently large uncertainty and thus fail to zoom in on promising solutions. Moreover, for the popular Matérn kernels, the covariance under the prior is essentially zero for two points if they differ substantially in one coordinate only. The use of trust regions results in more exploitation and often a better fit for the local surrogate. `SCBO` maintains the invariant that the trust region is centered at a point of maximum utility. Thus, the trust region is moved through the domain Ω as better points are discovered.

The generalization to black-box constraints poses additional fundamental problems that were not considered by Eriksson et al. [2019]. For many problems it is hard to even find a feasible solution, since the feasible set is typically non-convex. An investigation in Sect. 4 demonstrates the difficulty of this task. Moreover, black-box functions often vary drastically in their characteristics across Ω. We will provide examples where some constraints exhibit a huge variability whereas others are smooth. `SCBO` applies tailored *transformations* that account for the specific roles of the objective and the constraints.

Extending Thompson sampling to constrained optimization. `SCBO` extends Thompson sampling (`TS`) to black-box constraints, and is to the best of our knowledge the first to do so. `TS` scales to large batches at low computational cost and is at least as effective as `EI`, as we demonstrate below. To select a point for the next batch, `SCBO` samples r candidate points in Ω (see the supplement for details).

Let $x_1, \ldots, x_r$ be the sampled candidate points. Then `SCBO` samples a *realization* $(\hat{f}(x_i), \hat{c}_1(x_i), \ldots, \hat{c}_m(x_i))^T$ for all x_i with $1 \leq i \leq r$ from the respective posterior distributions on the functions $f, c_1, \ldots, c_m$. Let $\hat{F} = \{x_i \mid \hat{c}_\ell(x_i) \leq 0 \text{ for } 1 \leq \ell \leq m\}$ be the set of points whose realizations are feasible. If $\hat{F} \neq \emptyset$ holds, `SCBO` selects an $\operatorname{argmin}_{x \in \hat{F}} \hat{f}(x)$. Otherwise `SCBO` selects a point of *minimum total violation* $\sum_{\ell=1}^m \max\{\hat{c}_\ell(x), 0\}$, breaking ties via the sampled objective value. While we found that this natural selection criterion is able to identify a feasible point quickly for smooth constraints, we observed that it struggles when functions vary significantly in their magnitudes.

Transformations of objective and constraints. The key observations are that for the objective function we are particularly interested in the locations of possible optima, whereas for constraints we are interested in identifying feasible areas, i.e., where the constraint function values become negative. Thus, we apply transformations that emphasize these areas particularly; see also Fig. 1. To the objective function, we apply a *Gaussian copula* [e.g., see Wilson and Ghahramani, 2010]. The Gaussian copula first maps all observations under the objective to quantiles using the empirical CDF. Then it maps the quantiles through an inverse Gaussian CDF. Note that this procedure magnifies differences between values that are at the end of the

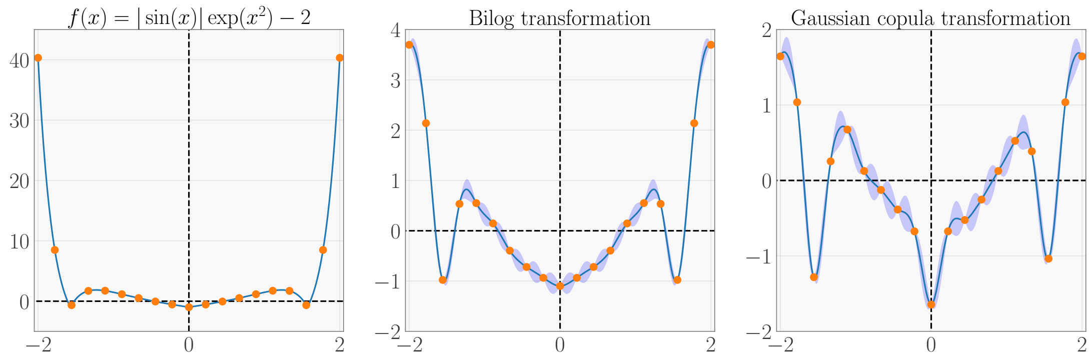

Figure 1: **(Left)** The original function where the distance to the origin varies considerably for the observations. If this was a constraint, the feasible region, denoted by the change of the sign, would be hard to detect. If it was the objective function, we would struggle to identify the minima, since the observations in the center differ only slightly and are considerably smaller in absolute value than the observations on the boundary. **(Middle)** The bilog transformation stretches out observations around zero, thereby making it easier to detect feasible areas. Note that a GP has been fitted to the observations given by the orange points in the middle and the right plot. The blue line depicts the posterior mean and the shaded area gives the posterior uncertainty of the GP. **(Right)** The copula transformation magnifies values that are at the ends of the observed spectrum, which facilitates the task of finding optima. Note that these transformations are advantageous over a naive standardization of each function as the latter is insensitive to the areas of interest.

observed range, i.e., minima or maxima. It affects the observed values but not their location. Finally, we apply Gaussian process regression to the mapped observations, as usual. For the constraints we employ the `bilog` transformation: $\text{bilog}(y) = \text{sgn}(y) \ln(1 + |y|)$ for a scalar observation y. It magnifies the range around zero to emphasize the change of sign that is decisive for feasibility. Moreover, it dampens large values.

Maintaining the trust region. The trust region is initialized as a hypercube with side length $L = L_{\text{init}}$. We count for each trust region the number of *successes* n_s and *failures* n_f since it was resized last. First suppose that all functions are observed without noise. Then a success occurs when `SCBO` observes a better point; by construction, this point must be inside the trust region. A failure happens when no point in the batch is better than the current center of the trust region. The center C of the trust region is chosen as follows. We select the best feasible point for C if any. Otherwise we pick a point with minimum total violation, again breaking ties via the objective. Note that we use (transformed) *observations* from the black-box functions, not *realizations* from the posterior. Thus, the center is moved to a new point whenever a success occurs. The trust region is resized as follows: if $n_s = \tau_s$ then the side length is set to $L = \min\{2L, L_{\max}\}$ and we reset $n_s = 0$. If $n_f = \tau_f$, then we set $L = L/2$ and $n_f = 0$. If the side length drops below a set threshold $L_{\min}$, then we initialize a new trust region. For noisy functions we follow the same rules, and use the posterior mean of GP model instead of the observed value. Note that the procedure for maintaining the trust regions follows [Eriksson et al., 2019] and is described here for completeness. In the next section we demonstrate that `SCBO` achieves excellent performance across all benchmarks.

3.1 Summary of the `SCBO` Algorithm

We summarize the `SCBO` algorithm.

1. Evaluate an initial set of points and initialize the trust region at a point of maximum utility.
2. Until the budget for samples is exhausted:
   (a) Fit GP models to the transformed observations.
   (b) Generate r candidate points $x_1, \ldots, x_r \in \Omega$ in the trust region.
   (c) For each of the q points of the next batch we sample a realization $\{(\hat{f}(x_i), \hat{c}_1(x_i), \ldots, \hat{c}_m(x_i))^T \mid 1 \le i \le r\}$ from the posterior over each candidate and add a point of maximum utility to the batch.
   (d) Evaluate the objective and constraints at the q new points.
   (e) Adapt the trust region by moving the center as described above. Update the counters n_s, n_f, and size L. If $L < L_{\min}$, initialize a new trust region.
3. Recommend an optimal feasible point (if any).

For noisy functions, we recommend a point of minimum posterior mean under all points that are feasible with probability at least $1-\delta$ (if any). Note that `SCBO` is *consistent* and hence will converge to a global optimum as the number of samples tends to infinity. The proof was deferred to the supplement due to space constraints.

4 Experimental Evaluation

We compare `SCBO` to the state-of-the-art: `PESC` [Hernández-Lobato et al., 2016] in `Spearmint`, `cEI` [Schonlau et al., 1998, Gardner et al., 2014], `SLACK` [Picheny et al., 2016] in `laGP`, the implementation of Jones et al. [2014] for `COBYLA` [Powell, 1994], `CMA-ES` [Hansen, 2006] in `pycma`, and random search (`RS`). Please see Sect. 1.1 for a discussion of these methods.

The Benchmarks. We evaluate the algorithms on a comprehensive selection of benchmark problems. First, we consider four low-dimensional problems in Sect. 4.1: a 3D tension-compression string problem with four constraints, a 4D pressure vessel design with with four constraints, a 4D welded beam design problem with five constraints, and a 7D speed reducer problem with eleven constraints. Next we consider the 10D Ackley problem with two constraints in Sect. 4.2 that is particularly interesting because of its small feasible region. Then we study four large-scale problems: the 30D Keane bump function with two constraints in Sect. 4.3, a 12D robust multi-point optimization problem with a varying number of constraints in Sect. 4.4, a 60D trajectory planning problem with 15 constraints in Sect. 4.5, and a 124D vehicle design problem with 68 constraints in Sect. 4.6. `PESC` and `SLACK` do not scale to large-scale high-dimensional problems and large batch sizes and are therefore omitted for these problems. Note that all benchmarks have multi-modal objective functions and are observed without noise. We perform 30 replications for each experiment.

To compare feasible and infeasible solutions, we adopt the rationale of Hernández-Lobato et al. [2016] that any feasible solution is preferable over an infeasible one and thus assign a default value to infeasible solutions that is set to the largest found objective value for the respective benchmark. Performance plots show the mean with one standard error. All methods start with an initial set of points given by a Latin hypercube design (LHD). `CMA-ES` and `COBYLA` are initialized from the best point in this design. Recall that `SCBO` applies transformations to the functions. In the supplement we investigate the performances of the baselines under these transformations and show that `SCBO` performs best.

4.1 Physics Test Problems

We evaluate the algorithms on a variety of physics problems. We use a budget of 100 evaluations, batch size $q = 1$, and 10 initial points. Fig. 2 summarizes the results for the four test problems. `SCBO` outperforms all baselines on the 3D tension-compression string problem [Hedar and Fukushima, 2006]: it found feasible solutions in all runs and consistently obtained excellent solutions. `PESC` and `cEI` are not competitive. Their performance is only slightly better than `RS` search on this problem. For the 4D pressure vessel design problem [Coello and Montes, 2002], `SCBO` obtains the best solutions followed by `cEI`, `PESC`, and `COBYLA`. `SCBO` also performs best for the 4D welded beam design problem [Hedar and Fukushima, 2006], followed by `cEI`. `SCBO` and `PESC` obtain excellent results for the 7D speed reducer design problem [Lemonge et al., 2010].

4.2 The 10D Ackley Function

We study the performance on the 10D Ackley function on the domain $[-5, 10]^{10}$ with the constraints $c_1(x) = \sum_{i=1}^{10} x_i \leq 0$ and $c_2(x) = \|x\|_2 - 5 \leq 0$. The Ackley function has a global optimum with value zero at the origin. This is a challenging problem where the probability of randomly selecting a feasible point is only $2.2 \cdot 10^{-5}$. We use a budget of 200 function evaluations, batch size $q = 1$, and 10 initial points. Fig. 3 shows that `COBYLA` initially makes good progress but is eventually outperformed by `SCBO` which achieves the best performance. `PESC` performs well, but is computationally costly: a run with `PESC` took 3 hours, while the other methods ran in minutes.

4.3 The 30D Keane Bump Function

The Keane bump function is a common test function for constrained global optimization [Keane, 1994]. This function has two constraints. We consider $d = 30$ and batch size 50 for `SCBO`, `cEI`, and `CMA-ES`. Each method uses 100 initial points. `COBYLA` does not support batching samples and thus samples sequentially, which is an advantage as it can leverage more data for acquisition. However, Fig. 3 shows that nonetheless `COBYLA` is not competitive. We see that `SCBO` clearly outperforms the other algorithms for this challenging high-dimensional benchmark. As stated above, we cannot compare to `PESC` and `SLACK` on this large-scale benchmark due to their computational overhead.

4.4 Robust Multi-point Optimization

Multi-point optimization is an important task in aerospace engineering [Liem et al., 2014, 2017, Martins, 2018]. Here, a design is optimized over a col-

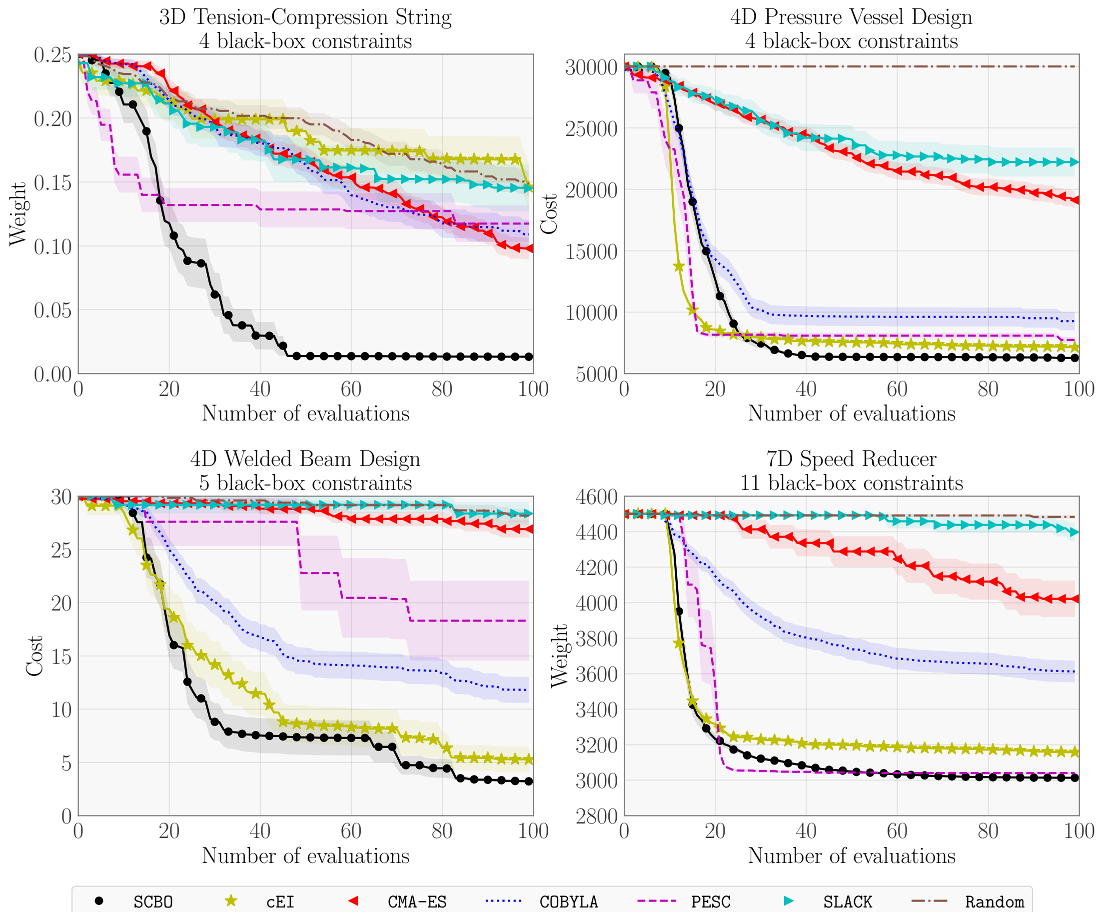

Figure 2: **(Upper left)** `SCBO` outperforms the other methods on the Tension-compression string problem. **(Upper right)** `SCBO` finds the best solutions on the pressure vessel design problem, followed by `cEI`, `PESC`, and `COBYLA`. **(Lower left)** `SCBO` performs best on the welded beam design problem. **(Lower right)** `SCBO` and `PESC` perform the best on the speed reducer problem.

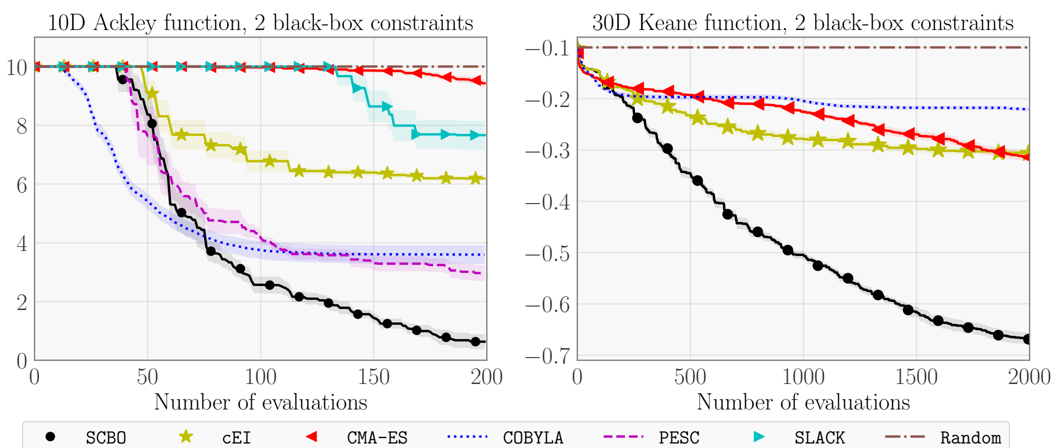

Figure 3: **(Left)** 10D Ackley function with two constraints. `SCBO` consistently finds solutions close to the global optimum. **(Right)** 30D Keane function with two constraints. `SCBO` clearly outperforms the other methods from the start.

lection of flight conditions. Multi-point optimization produces designs with better practical performance by addressing the issue that tuning a design for a single scenario often leads to designs with poor off-scenario performance [Jameson, 1990, Cliff et al., 2001]. In this section we propose a *robust multi-point optimization problem*. The goal is to optimize the performance of the design x averaged over m scenarios (potentially weighted by importance), subject to individual constraints that assert an acceptable performance for each scenario. Our problem is derived from the lunar lander problem, where the goal is to find a 12D controller that maximizes the reward averaged over m terrains. We extend this problem by adding m constraints that assert that no individual reward is below 200, which guarantees that the lunar lands successfully. Without these constraints, the algorithms often produce policies that occasionally crash the lander. We evaluate the algorithms with 1000 samples, batch size $q = 50$, and 50 initial points for three experiments that differ in the number of constraints: $m = 10$, $m = 30$, and $m = 50$. Tab. 1 summarizes the results.

m		SCBO	cEI	CMA-ES	COBYLA	RS
10	Best	321.3	322.2	310.3	315.0	NA
	Worst	302.8	250.9	266.4	293.0	NA
	Median	318.0	318.5	291.9	299.7	NA
	Feasible	28/30	27/30	21/30	2/30	0/30
30	Best	316.2	311.2	293.2	312.7	NA
	Worst	295.2	267.2	270.3	294.2	NA
	Median	311.8	288.4	283.7	295.2	NA
	Feasible	26/30	10/30	11/30	5/30	0/30
50	Best	309.7	295.4	295.7	300.0	NA
	Worst	276.2	262.7	256.3	276.5	NA
	Median	306.0	269.9	274.5	285.1	NA
	Feasible	27/30	8/30	11/30	3/30	0/30

Table 1: Results for the 12D multi-point optimization problem. We observe that SCBO finds the best robust policies over thirty runs and scales best to larger numbers of constraints.

We see that SCBO found a feasible controller for 28/30 runs when $m = 10$. cEI and CMA-ES found feasible points in 27/30 and 21/30 runs respectively. COBYLA and RS struggled visibly. We note that SCBO clearly outperformed the other methods for the more constrained settings with $m = 30$ and $m = 50$. Note that the best reward found by each algorithm clearly decreases when m increases. The problem becomes harder when we add more scenarios, since feasible solutions for $m = 10$ may not satisfy all constraints for $m = 30$ or $m = 50$.

4.5 60D Rover Trajectory Planning

We study a 60D route planning problem adapted from [Wang et al., 2018]. The task is to position 30 waypoints that lead a rover on a path of minimum cost from its starting position to its destination, while avoiding collisions with obstacles. We propose a constraint-based extension with $m = 15$ constraints that are met if and only if the rover does not collide with any associated impassable obstacles. The exact formulation of these constraints is given in the supplementary material. Fig. 4 (middle) illustrates the setup and the best trajectory found by SCBO. There are two types of terrain that vary in their cost: the green terrain can be traversed at cost zero and a yellow terrain that inflicts a certain cost. This problem turns out to be challenging for small sampling budgets. Thus, we have evaluated SCBO, cEI, CMA-ES, COBYLA, and RS for a total of 5000 evaluations with batch size $q = 100$ and 100 initial points. Fig. 4 (left) summarizes the performances. We see that SCBO outperforms the other methods by far on this hard benchmark.

4.6 124D Vehicle Design with 68 Constraints

We evaluate the algorithms on a 124-dimensional vehicle design problem MOPTA08 [Anjos, 2008], where the goal is to minimize the mass of a vehicle subject to 68 performance constraints. The 124 variables describe gages, materials, and shapes. We ran all experiments with a budget of 2000 samples, batch size q=10, and 130 initial points. We point out that this benchmark showcases the scalability of the implementation of SCBO that uses GPyTorch [Gardner et al., 2018] and KeOPS [Charlier et al., 2018] to fit the 69 GP models in a batch; see the supplement for details. Fig. 4 (right) shows SCBO, cEI, CMA-ES, and COBYLA over 30 runs. SCBO found a feasible solution in all 30 runs and the best solution found by SCBO had value 236.7. COBYLA found a feasible point in only 11/30 runs, one which had objective value 238.8, while cEI was not able to find a feasible solution with function value below 300.

4.7 Ablation studies

We investigate how the various components in SCBO contribute to the overall performance: specifically, how does the application of i) the transformations (Transformed, Untransformed), ii) the acquisition criterion (TS or EI), and iii) the use of a trust region (TR, Global) affect the performance. Fig. 5 summarizes the performances of all eight combinations on three benchmarks. On the left, we see that the use of the trust region is critical for the 30D Keane function. Approaches that do not use a trust region struggle, just as in Sect. 4.6. Moreover, the transformation provides an additional gain, whereas the choice of the acquisition function has no noticeable effect. The center plot is for the 2D toy problem proposed by Hernández-Lobato et al. [2016] that has a smooth objective function and two easy

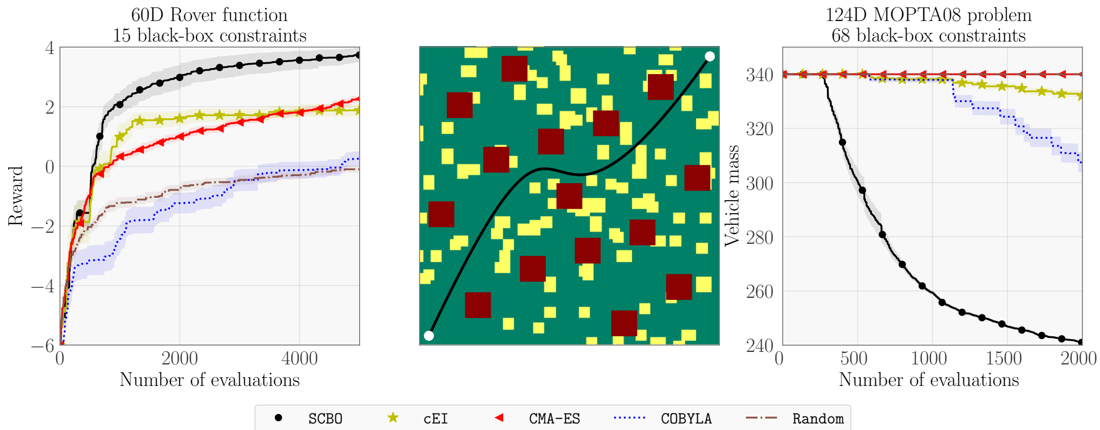

Figure 4: **(Left) 60D trajectory planning with 15 constraints:** `SCBO` finds excellent solutions quickly and outperforms the other methods. **(Middle) Illustration of the trajectory planning problem**: The black line is the best trajectory found by `SCBO` with a reward of 4.93. The green area can be traversed at no cost. Yellow squares denote terrain that inflicts a cost upon traversal. Red squares are impassable obstacles. **(Right) 124D Vehicle Design with 68 Constraints**: `SCBO` finds a feasible point in 30/30 runs and consistently finds good solutions.

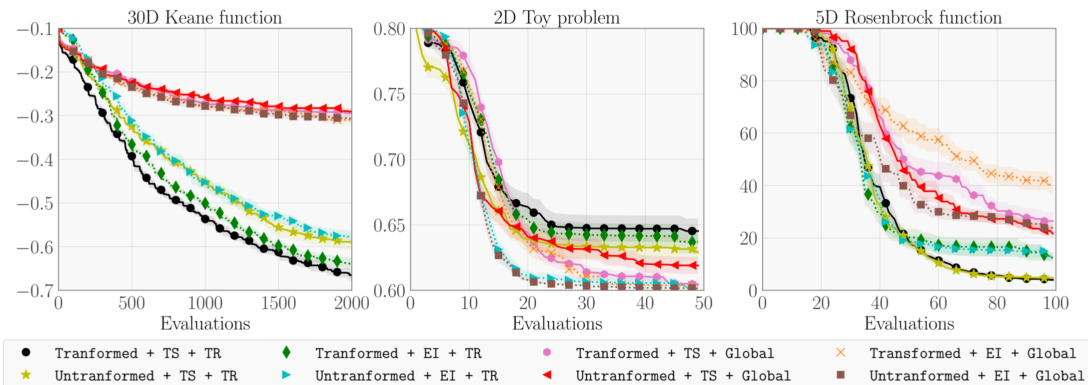

Figure 5: We investigate the effects of (i) the transformations, (ii) different acquisition functions (TS/EI), and (iii) the trust region (TR). **(Left)** 30D Keane function with 2 black-box constraints. **(Middle)** 2D Toy problem with 2 black-box constraints of Hernández-Lobato et al. [2016]. **(Right)** 5D Rosenbrock function with 2 poorly scaled black-box constraints.

constraints. Here, BO without a trust region and `EI` should shine, and this is indeed the case. The right plot considers the 5D Rosenbrock function with two poorly scaled constraints. Again the trust region is critical for a good performance, as fitting a global surrogate model seems challenging. Interestingly, `TS` achieves significantly better results than `EI`.

5 Conclusions

We studied the task of optimizing a black-box objective function under black-box constraints that has numerous applications in machine learning, control, and engineering. We found that the existing methods struggle in the face of multiple constraints and more than just a few decision variables. Therefore, we proposed the *Scalable Constrained Bayesian Optimization* (`SCBO`) algorithm that leverages tailored transformations of the underlying functions together with the trust region approach of Eriksson et al. [2019] and Thompson sampling (TS) to scale to high-dimensional spaces and large sampling budgets.

We performed a comprehensive experimental evaluation that compared `SCBO` to the state-of-the-art from machine learning, operations research, and evolutionary algorithms on a variety of benchmark problems that span control, multi-point optimization, and physics. We found that `SCBO` outperforms the state-of-the-art on high-dimensional benchmarks, and matches or beats the performance of the best baseline otherwise. In the supplement, we also provide an efficient GPU implementation of SCBO based on batch-GPs and a formal proof that SCBO converges to a global optimum.

For future work, we are interested in applications where the objective and constraints have substantial correlations. For example, consider the design of an aircraft wing: here the aerodynamic performance (e.g., lift and drag), the structural stability, and the fuel-burn will be related. If the airfoil's geometry generates turbulent structures, the drag will increase and the fuel burn will suffer. The heterogeneity of the involved functions may make the adoption of a multi-output Gaussian process challenging. We believe that leveraging these correlations may pave an avenue towards solving problems with hundreds of constraints more efficiently. Constraints also arise naturally for combinatorial black-box functions [Baptista and Poloczek, 2018, Oh et al., 2019] that have exciting applications in engineering and science.

Moreover, we look forward to inter-disciplinary applications: SCBO's ability to optimize high-dimensional constrained problems will allow to optimize an airfoil described by a mesh or the parameters controlling a chemical process, e.g., for growing nanotubes or when searching for a solar cell material [Herbol et al., 2018, Ortoll-Bloch et al., 2019].

References

M. Anjos. The MOPTA 2008 benchmark, 2008. URL `http://www.miguelanjos.com/jones-benchmark`.

S. Ariafar, J. Coll-Font, D. Brooks, and J. Dy. ADMMBO: Bayesian optimization with unknown constraints using ADMM. *Journal of Machine Learning Research*, 20(123):1–26, 2019.

D. V. Arnold and N. Hansen. A (1+ 1)-CMA-ES for constrained optimisation. In *Proceedings of the 14th annual conference on Genetic and evolutionary computation*, pages 297–304. ACM, 2012.

R. Baptista and M. Poloczek. Bayesian optimization of combinatorial structures. In *International Conference on Machine Learning*, pages 462–471, 2018.

M. Binois, D. Ginsbourger, and O. Roustant. A warped kernel improving robustness in Bayesian optimization via random embeddings. In *International Conference on Learning and Intelligent Optimization*, pages 281–286. Springer, 2015.

M. Binois, D. Ginsbourger, and O. Roustant. On the choice of the low-dimensional domain for global optimization via random embeddings. *Journal of global optimization*, 76(1):69–90, 2020.

M. A. Bouhlel, N. Bartoli, R. G. Regis, A. Otsmane, and J. Morlier. Efficient global optimization for high-dimensional constrained problems by using the kriging models combined with the partial least squares method. *Engineering Optimization*, 50(12):2038–2053, 2018.

B. Charlier, J. Feydy, and J. A. Glaunés. KeOps, 2018. URL `https://github.com/getkeops/keops`.

S. E. Cliff, J. J. Reuther, D. A. Saunders, and R. M. Hicks. Single-point and multipoint aerodynamic shape optimization of high-speed civil transport. *Journal of Aircraft*, 38(6):997–1005, 2001.

C. A. C. Coello and E. M. Montes. Constraint-handling in genetic algorithms through the use of dominance-based tournament selection. *Advanced Engineering Informatics*, 16(3):193–203, 2002.

D. Eriksson, K. Dong, E. Lee, D. Bindel, and A. G. Wilson. Scaling Gaussian process regression with derivatives. In *Advances in Neural Information Processing Systems*, pages 6867–6877, 2018.

D. Eriksson, M. Pearce, J. Gardner, R. D. Turner, and M. Poloczek. Scalable global optimization via local Bayesian optimization. In *Advances in Neural Information Processing Systems 32*, pages 5497–5508. 2019.

A. Forrester, A. Sobester, and A. Keane. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.

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

J. Gardner, G. Pleiss, K. Q. Weinberger, D. Bindel, and A. G. Wilson. GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. In *Advances in Neural Information Processing Systems*, pages 7576–7586, 2018.

J. R. Gardner, M. J. Kusner, Z. E. Xu, K. Q. Weinberger, and J. P. Cunningham. Bayesian optimization with inequality constraints. In *ICML*, pages 937–945, 2014.

M. A. Gelbart, J. Snoek, and R. P. Adams. Bayesian optimization with unknown constraints. In *30th Conference on Uncertainty in Artificial Intelligence, UAI 2014*, pages 250–259. AUAI Press, 2014.

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

N. Hansen. The CMA evolution strategy: A comparing review. In *Towards a New Evolutionary Computation*, pages 75–102. Springer, 2006. Code available at: `https://github.com/CMA-ES/pycma`. Last accessed on 02/03/2020.

A.-R. Hedar and M. Fukushima. Derivative-free filter simulated annealing method for constrained continuous global optimization. *Journal of Global optimization*, 35(4):521–549, 2006.

H. C. Herbol, W. Hu, P. Frazier, P. Clancy, and M. Poloczek. Efficient search of compositional space for hybrid organic–inorganic perovskites via Bayesian optimization. *npj Computational Materials*, 4(1):1–7, 2018.

J. M. Hernández-Lobato, M. W. Hoffman, and Z. Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in Neural Information Processing Systems*, pages 918–926, 2014.

J. M. Hernández-Lobato, M. A. Gelbart, R. P. Adams, M. W. Hoffman, and Z. Ghahramani. A general framework for constrained Bayesian optimization using information-based search. *The Journal of Machine Learning Research*, 17(1):5549–5601, 2016. Code available at: `https://github.com/HIPS/Spearmint/tree/PESC`. Last accessed on 02/03/2020.

J. M. Hernández-Lobato, J. Requeima, E. O. Pyzer-Knapp, and A. Aspuru-Guzik. Parallel and distributed Thompson sampling for large-scale accelerated exploration of chemical space. In *Proceedings of the International Conference on Machine Learning*, pages 1470–1479, 2017.

A. Jameson. Automatic design of transonic airfoils to reduce the shock induced pressure drag. In *Proceedings of the 31st Israel annual conference on aviation and aeronautics, Tel Aviv*, pages 5–17, 1990.

E. Jones, T. Oliphant, and P. Peterson. SciPy: Open source scientific tools for Python. 2014.

K. Kandasamy, A. Krishnamurthy, J. Schneider, and B. Póczos. Parallelised Bayesian optimisation via Thompson sampling. In *International Conference on Artificial Intelligence and Statistics*, pages 133–142, 2018.

A. Keane. Experiences with optimizers in structural design. In *Proceedings of the conference on adaptive computing in engineering design and control*, volume 94, pages 14–27, 1994.

J. Kirschner, M. Mutny, N. Hiller, R. Ischebeck, and A. Krause. Adaptive and safe Bayesian optimization in high dimensions via one-dimensional subspaces. In *International Conference on Machine Learning*, pages 3429–3438, 2019.

O. Kramer. A review of constraint-handling techniques for evolution strategies. *Applied Computational Intelligence and Soft Computing*, 2010, 2010.

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

R. Lam, M. Poloczek, P. Frazier, and K. E. Willcox. Advances in Bayesian optimization with applications in aerospace engineering. In *2018 AIAA Non-Deterministic Approaches Conference*, page 1656, 2018.

R. R. Lam and K. E. Willcox. Lookahead Bayesian optimization with inequality constraints. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1888–1898, 2017.

A. Lemonge, H. Barbosa, C. Borges, and F. Silva. Constrained optimization problems in mechanical engineering design using a real-coded steady-state genetic algorithm. *Mecánica Computacional*, 29:9287–9303, 2010.

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

B. Letham, R. Calandra, A. Rai, and E. Bakshy. Re-examining linear embeddings for high-dimensional Bayesian optimization. *Advances in Neural Information Processing Systems*, 33, 2020.

R. P. Liem, G. K. Kenway, and J. R. Martins. Multimission aircraft fuel-burn minimization via multipoint aerostructural optimization. *AIAA Journal*, 53(1): 104–122, 2014.

R. P. Liem, J. R. R. A. Martins, and G. K. Kenway. Expected drag minimization for aerodynamic design optimization based on aircraft operational data. *Aerospace Science and Technology*, 63:344–362, 2017.

J. R. R. A. Martins. Personal communication, 2018.

L. Mathesen, G. Pedrielli, S. H. Ng, and Z. B. Zabinsky. Stochastic optimization with adaptive restart: A framework for integrated local and global learning. *Journal of Global Optimization*, pages 1–24, 2020.

M. Mutny and A. Krause. Efficient high dimensional Bayesian optimization with additivity and quadrature Fourier features. In *Advances in Neural Information Processing Systems*, pages 9005–9016, 2018.

A. Nayebi, A. Munteanu, and M. Poloczek. A framework for Bayesian optimization in embedded subspaces. In *International Conference on Machine Learning*, pages 4752–4761, 2019. The code is available at `https://github.com/aminnayebi/HesBO` and in `https://github.com/wujian16/Cornell-MOE`. Last accessed on 02/03/2020.

C. Oh, E. Gavves, and M. Welling. BOCK : Bayesian optimization with cylindrical kernels. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 3868–3877, 2018.

C. Oh, J. Tomczak, E. Gavves, and M. Welling. Combinatorial Bayesian optimization using the graph Cartesian product. In *Advances in Neural Information Processing Systems 32*, pages 2910–2920. 2019.

A. G. Ortoll-Bloch, H. C. Herbol, B. A. Sorenson, M. Poloczek, L. A. Estroff, and P. Clancy. Bypassing solid-state intermediates by solvent engineering the crystallization pathway in hybrid organic–inorganic perovskites. *Crystal Growth & Design*, 2019.

J. M. Parr, A. I. Forrester, A. J. Keane, and C. M. Holden. Enhancing infill sampling criteria for surrogate-based constrained optimization. *Journal of Computational Methods in Sciences and Engineering*, 12(1-2):25–45, 2012a.

J. M. Parr, A. J. Keane, A. I. Forrester, and C. M. Holden. Infill sampling criteria for surrogate-based optimization with constraint handling. *Engineering Optimization*, 44(10):1147–1166, 2012b.

V. Picheny. A stepwise uncertainty reduction approach to constrained global optimization. In *Artificial Intelligence and Statistics*, pages 787–795, 2014.

V. Picheny, R. B. Gramacy, S. Wild, and S. Le Digabel. Bayesian optimization under mixed constraints with a slack-variable augmented Lagrangian. In *Advances in neural information processing systems (NIPS)*, pages 1435–1443, 2016. Code available at: `https://cran.r-project.org/web/packages/laGP`. Last accessed on 02/03/2020.

M. J. Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in Optimization and Numerical Analysis*, pages 51–67. Springer, 1994.

M. J. Powell. A view of algorithms for optimization without derivatives. *Mathematics Today-Bulletin of the Institute of Mathematics and its Applications*, 43 (5):170–174, 2007.

W. B. Powell. A unified framework for stochastic optimization. *European Journal of Operational Research*, 275(3):795–821, 2019.

P. Rolland, J. Scarlett, I. Bogunovic, and V. Cevher. High-dimensional Bayesian optimization via additive models with overlapping groups. In *International Conference on Artificial Intelligence and Statistics*, pages 298–307, 2018.

D. Salinas, H. Shen, and V. Perrone. A copula approach for hyperparameter transfer learning. *arXiv preprint arXiv:1909.13595*, 2019.

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

B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016.

E. Snelson, C. E. Rasmussen, and Z. Ghahramani. Warped gaussian processes. *Advances in neural information processing systems*, 16:337–344, 2004.

J. Snoek, K. Swersky, R. Zemel, and R. Adams. Input warping for bayesian optimization of non-stationary functions. In *International Conference on Machine Learning*, pages 1674–1682. PMLR, 2014.

A. Sóbester, A. I. Forrester, D. J. Toal, E. Tresidder, and S. Tucker. Engineering design applications of surrogate-assisted optimization techniques. *Optimization and Engineering*, 15(1):243–265, 2014.

W. R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

Z. Wang, F. Hutter, M. Zoghi, D. Matheson, and N. de Freitas. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, 55:361–387, 2016.

Z. Wang, C. Gehring, P. Kohli, and S. Jegelka. Batched large-scale Bayesian optimization in high-dimensional spaces. In *International Conference on Artificial Intelligence and Statistics*, pages 745–754, 2018.

A. G. Wilson and Z. Ghahramani. Copula processes. In *Advances in Neural Information Processing Systems*, pages 2460–2468, 2010.