

Self-optimizing vapor compression cycles online with Bayesian optimization under local search region constraints

Joel A. Paulson

Farshud Sorourifar

William G. Lowrie Department of Chemical and Biomolecular Engineering
The Ohio State University
Columbus, OH 43210

Email: {paulson.82@osu.edu, sorourifar.1@buckeyemail.osu.edu}

Christopher R. Laughman

Ankush Chakrabarty

Multiphysical Systems (MS) Team
Mitsubishi Electric Research Laboratories (MERL)
Cambridge, MA 02139

Email: {laughman, chakrabarty}@merl.com

Self-optimizing efficiency of vapor compression cycles (VCCs) involves assigning multiple decision variables simultaneously in order to minimize power consumption while maintaining safe operating conditions. Due to the modeling complexity associated with cycle dynamics (and other smart building energy systems), online self-optimization requires algorithms that can safely and efficiently explore the search space in a derivative-free and model-agnostic manner. This makes Bayesian optimization (BO) a strong candidate for self-optimization. Unfortunately, classical BO algorithms ignore the relationship between consecutive optimizer candidates, resulting in jumps in the search space that can lead to fail-safe mechanisms being triggered, or undesired transient dynamics that violate operational constraints. To this end, we propose safe LSR-BO, a global optimization methodology that builds on the BO framework while enforcing two types of safety constraints including black-box constraints on the output and local search region (LSR) constraints on the input. We provide theoretical guarantees that under standard assumptions on the performance and constraint functions, LSR-BO guarantees constraints will be satisfied at all iterations with high probability. Furthermore, in the presence of only input LSR constraints, we show the method will converge to the true (unknown) globally optimal solution. We demonstrate the potential of our proposed LSR-BO method on a high-fidelity simulation model of a commercial vapor compression system with both LSR constraints on expansion valve positions and fan speeds, in addition to other safety constraints on discharge and evaporator temperatures.

Nomenclature

BO Bayesian optimization

DFO Derivative-free optimization

EI Expected improvement

EIC Expected improvement with constraints

GP Gaussian process

GPR Gaussian process regression

HVAC Heating, ventilation, and cooling

LSR Local search region

LSR-BO LSR-constrained BO (pronounced ‘laser-BO’)

NEB No-empty-ball property

VCC Vapor compression cycle

VCS Vapor compression system

1 Introduction

Vapor compression cycles (VCCs) are a critical component in many energy transfer systems due to their reliability, especially in heat pumps and refrigeration [1]. Most commonly, these VCCs comprise of: compressors for pumping and circulating fluids, expansion valves for regulating pressures in different parts of the cycle, and heat exchangers for thermal energy transfer from one medium to another. Actuator variables associated with each of these components contribute significantly to cycle performance, and therefore, energy efficiency. Two popular methods for designing control policies for these actuators are through the use of decentralized feedback loops with proportional-integral controllers [2] and model predictive controllers [3]. Often, these controllers are ‘legacy’: that is, their gains are pre-fixed by the manufacturer, and therefore, the only degree of freedom afforded for online optimization is by adjusting outer loop parameters such as setpoints to the legacy controllers. Selecting good setpoints unfortunately introduces additional complex-

ity into the design. Proper assignment of actuator setpoints is a well-understood approach for optimizing closed-loop performance, despite having legacy feedback controllers already deployed in the system by the manufacturer [4] and other sources of uncertainties and disturbances arising in practice: such as disturbances in ambient conditions, human occupancy levels, end-user heat load variations. This is further exacerbated by the fact that the closed-loop system dynamics are nonlinear, multi-variable, and often contain unmodeled components.

To bring this problem into an optimization context, we reiterate that closed-loop systems can be further optimized after deployment by adapting decision variables given some useful performance metric defined in terms of measured operational data. Manually tuning these variables can be highly inefficient or require considerable task-specific expertise that does not generalize to new types of systems. We posit that auto-tuning methods are a systematic and efficient approach to attack this problem. Auto-tuning algorithms are capable of automatically adjusting the control parameters to achieve optimal performance as a way to save time, manual effort, and cost of experimentation; see for instance [5–8] for some examples where auto-tuning has been reported to improve the performance of modern engineering systems.

Since the map between the control parameters and relevant closed-loop performance functions are often unmodeled (unknown) and may be highly nonlinear, it is common to treat this map as a black-box function with fully unknown structure. Therefore, we can treat auto-tuning as a black-box optimization problem, which can be tackled using any derivative-free optimization (DFO) method; we refer the reader to [9] for a detailed overview of DFO algorithms. Since closed-loop experimentation or high-fidelity software simulations (like so-called digital twins) are needed to accurately represent true system behavior, the evaluation of which is expensive, auto-tuning algorithms must be designed to require as few experiments/evaluations as possible.

Bayesian optimization (BO) is a *sample-efficient* DFO method that uses a probabilistic machine learning model to intelligently search over feasible parameter spaces [10, 11]. Due to its sample efficiency, BO has received significant attention in the context of auto-tuning including real-world applications in wind energy systems [12], engines [13], and space cooling [14, 15]. In addition, BO has been shown to generalize well across a wide-variety of complex control problems such as controller parameter/reference setpoint tuning [16–18], cascade controller tuning [19], and MPC tuning [20–23]. An important challenge in the online deployment of such traditional BO-based auto-tuning methods is that they do not directly consider any form of safety. In the energy optimization tasks of interest here, these safety constraints can come in two forms: (i) constraints on the allowable closed-loop output variables and (ii) constraints on the movement of the input controller design parameters. An example of (i) is any outcome that can lead to major economic or social consequences such as the discharge or evaporator temperatures exceeding known limits that could lead to failure of the VCC system. Constraints of the form (ii)

are more related to aggressive changes in the inputs. For example, energy consumption in HVAC systems is correlated with the electronic expansion valve (EEV) position setpoint. Aggressively exploring the search space is tantamount to aggressively opening and closing the EEV, which causes excessive wear and tear of the mechanical components of the valve and can produce oscillations in the refrigerant flow dynamics that result in compromised heating/cooling performance and can destabilize feedback loops that regulate the behavior of other system actuators (e.g., compressor speed).

Recent work on so-called safe BO, e.g., [24, 25] has considered black-box output constraints; however, these methods explore the search space without considering the distance between consecutive optimization candidates. As such, they have the tendency to suggest candidates at consecutive iterations that are far apart in the search space, which leads to the aggressive dynamic behavior described above. We recently developed a novel BO extension, called LSR-BO, that enforces local search region (LSR) constraints to restrict how much the controller tuning parameters can be varied at each iteration as a way to address this issue [26]. However, our prior work only considers a single closed-loop performance indicator and thus cannot handle black-box safety-critical output constraints and assumes perfect measurements so it does not directly handle noisy observations. Therefore, the main goal of this paper is to overcome these limitations by extending LSR-BO to handle critical closed-loop safety constraints and noisy observations in the performance and constraint functions. The proposed extension, safe LSR-BO, is conceptually simple and computationally efficient, making it easy to implement in practice. Furthermore, we provide new theoretical analysis of safe LSR-BO to show it confers some useful safety and convergence properties.

In summary, our **major contributions** are:

- (i) extending our recently proposed LSR-BO method [26] to provide a high probability guarantee of closed-loop safety constraint satisfaction, while balancing local and global performance improvement;
- (ii) introducing an effective strategy for handling noisy observations of the performance and constraint functions;
- (iii) establishing that safe LSR-BO will generate a sequencing of candidate tuning parameters that lead to safe performance for all iterations (with high probability) and can converge to the true (unknown) globally optimal solution under certain conditions; and,
- (iv) demonstration of safe LSR-BO’s ability to ensure safe energy optimization of industrial heat pump systems using high-fidelity simulations.

The remainder of the paper is organized as follows. In Section 2, we describe the importance of the safe LSR-constrained BO problem, and provide a detailed overview of our proposed solution in Section 3. We perform some theoretical analysis on the safety and convergence properties of LSR-BO in Section 4. We then demonstrate the ability of safe LSR-BO to outperform existing BO algorithms on a benchmark and real-world energy application in Section 5. We present our conclusions in Section 6.

2 Self-Optimizing VCCs and the Criticality Of LSR Constraints

As explained in §1, vapor compression systems with legacy controllers in place can be treated as black-box closed-loop systems as the internal dynamics are not fully known for online optimization. Therefore, we consider closed-loop systems abstracted by

$$x_+ = F(x, \theta), \quad (1)$$

where $x \in \mathbb{R}^{n_x}$ and $x_+ \in \mathbb{R}^{n_x}$ denote the current and one-step future state of the vapor compression system, respectively, and $\theta \in \Theta \subset \mathbb{R}^d$ denotes a set of control-relevant parameters that can be assigned. More generally, the parameters θ can represent any tunable variable in the control policy, which includes parameter choices (e.g., controller gains) or structural choices (e.g., turning on or off a component of the policy), but in this work, θ comprises references (e.g., setpoints) for the legacy controllers to track.

To judge the efficiency of the cycle, we assume that some performance function $f^0 : \Theta \rightarrow \mathbb{R}$ can be defined; in this work, $f^0(\theta)$ quantifies the energy consumption for any given $\theta \in \Theta$ over a pre-defined time interval. Temperature constraints are often explicitly specified in addition to cycle performance to prevent unsafe operating conditions: therefore, we assume that a set of m safety constraints $f^i : \Theta \rightarrow \mathbb{R}$ can be defined such that $f^i(\theta) \geq 0$ for all $i = 1, \dots, m$ defines the safe operating region.

Consequently, the energy minimization task for the VCC with safety constraints is formulated as:

$$\theta^* \in \operatorname{argmax}_{\theta \in \Theta} \{f^0(\theta) : f^i(\theta) \geq 0\}, \quad (2)$$

where θ^* denotes the globally optimal control policy parameters. The characteristics of the constraint functions $\{f^i\}_{i=0}^m$ restrict the types of algorithms that can be employed to tackle (2). If the mathematical structure of these functions are known and they are convex, we can apply established methods from the field of convex optimization to efficiently identify θ^* . However, in the majority of real-world applications, the structure of the cost and constraint functions is often unknown and may be highly non-convex. This is especially true when one attempts to solve (2) using closed-loop data collected from experimental systems (such as VCCs) for which $f^0(\theta), \dots, f^m(\theta)$ can have a complex dependence on θ due to its interactions with several components (the system and actuator dynamics, the control policy, and the cost and constraint specifications).

Therefore, we tackle the closed-loop optimization problem (2) by considering $\{f^i\}_{i=0}^m$ to be purely *black-box* constraint functions. This implies that we must interactively learn about them through repeated queries at particular $\theta \in \Theta$ values. Specifically, we must design some sequence of N parameter values $\{\theta_1, \dots, \theta_N\}$ for which we can simultaneously obtain the performance and constraint values $\{\{f^i(\theta_1)\}_{i=0}^m, \dots, \{f^i(\theta_N)\}_{i=0}^m\}$. There has been a significant

amount of work on algorithms for generating this sequence of points, with the goal of driving $\theta_N \rightarrow \theta^*$ as quickly as possible, including Bayesian optimization (BO). The key advantage of BO is that it is specifically designed for expensive functions for which limited data can be generated. This is an important consideration in the applications of interest in this work since closed-loop experimentation often requires significant time and/or resources.

Although BO (and its variants) can be used to tackle (2), BO methods are designed for static objective problems and therefore, there is no restriction on the distance between consecutive optimizer candidates selected by the algorithms. In fact, BO commonly exhibits the tendency to produce consecutive candidates that are a large distance (w.r.t. some metric on Θ) apart. This is natural since BO wants to ensure sufficient exploration of Θ ; however, in most industrial applications, it is not possible to change θ by a large amount within a short span of time. In the context of VCCs (1), large changes in θ can induce aggressive dynamics causing safety constraint violation and/or triggering of fail-safe mechanism that limit performance quality. To prevent undesired dynamics due to large jumps during online optimization, our proposed method systematically restricts the consecutive candidates θ_n and θ_{n+1} to be within a domain-informed safe neighborhood of the current candidate θ_n .

Formally, this restriction can be written as

$$\theta_{n+1} \in \mathcal{B}_\delta(\theta_n) \quad \forall n \in \{0, \dots, N-1\}, \quad (3a)$$

$$\text{where, } \mathcal{B}_\delta(\theta) = \{\theta_+ \in \Theta : \|\theta_+ - \theta\|_p \leq \delta\}, \quad (3b)$$

denotes a p -norm ball of radius $\delta > 0$ centered at θ . We refer to the proposed constraints (3) as *local search region* (LSR) constraints. Traditional BO methods are not equipped to handle such restrictions on their exploration capabilities. Therefore, the main goal of this work is to develop a safe BO method for solving (2) that respects LSR constraints (3). We describe our proposed LSR-BO algorithm next, and establish performance certificates (in terms of safety and convergence) in the subsequent section.

3 A Practical Safe Bayesian Optimization Method with Local Search Constraints

In this section, we begin by providing a brief overview of classical BO and recent results on so-called “safe” BO that are equipped to satisfy constraints during online optimization (with high probability). With notation and parlance borrowed from those methods, we subsequently discuss how to handle LSR constraints while ensuring efficient online optimization via LSR-BO.

3.1 Classical Bayesian Optimization

Since the functions $\{f^i\}_{i=0}^m$ are unknown in (2), they must be learned from data collected from the VCC during online optimization experiments. Assuming these constraint

functions satisfy smoothness properties, they can be effectively modeled as independent Gaussian processes (GPs) of the form

$$f^i(\theta) \sim \text{GP}(\mu^i(\theta), k^i(\theta, \theta')), \quad \forall i = 0, \dots, m, \quad (4)$$

where $\mu^i(\theta) = \mathbb{E}\{f^i(\theta)\}$ denotes the prior mean function and $k^i(\theta, \theta') = \mathbb{E}\{(f^i(\theta) - \mu^i(\theta))(f^i(\theta') - \mu^i(\theta'))\}$ denotes the prior covariance (kernel) function. GP models are non-parametric and have the nice property that the posterior model, conditioned on noisy observations, remains a GP with updated mean and covariance functions. In particular, let

$$\mathcal{D}_n^i = \{(\theta_j, f_j^i(\theta))\}_{j=1}^n$$

denote the current set of n observations of function i , then posterior prediction of $f^i(\theta)$ at any future test point $\theta \in \Theta$ is given by

$$f^i(\theta) | \mathcal{D}_n \sim \mathcal{N}(\mu_n^i(\theta), k_n^i(\theta, \theta)), \quad (5a)$$

where

$$\mu_n^i(\theta) = \mu^i(\theta) + \mathbf{k}_n^{i\top}(\theta)(\mathbf{K}_n^i)^{-1}\tilde{\mathbf{f}}_n^i, \quad (5b)$$

$$k_n^i(\theta, \theta') = k^i(\theta, \theta') - \mathbf{k}_n^{i\top}(\theta)(\mathbf{K}_n^i)^{-1}\mathbf{k}_n^i(\theta'), \quad (5c)$$

with

$$\begin{aligned} \mathbf{k}_n^i(\theta) &= [k^i(\theta_1, \theta), \dots, k^i(\theta_n, \theta)]^\top && \in \mathbb{R}^{n \times 1}, \\ \tilde{\mathbf{f}}_n^i &= [f^i(\theta_1) - \mu^i(\theta_1), \dots, f^i(\theta_n) - \mu^i(\theta_n)]^\top && \in \mathbb{R}^{n \times 1}, \\ \mathbf{K}_n^i &= [k^i(\theta_i, \theta_j)]_{(i,j) \in \{1, \dots, n\} \times \{1, \dots, n\}} && \in \mathbb{R}^{n \times n}. \end{aligned}$$

The choice of the prior determines the properties of the fitted functions and plays an important role in the accuracy of the GP. We will focus on covariance functions belonging to the Matérn class that have a parameter ν that controls the degree of smoothness of the performance function, i.e.,

$$k_\nu^i(\theta, \theta') = \zeta^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} d(\theta, \theta') \right) B_\nu \left(\sqrt{2\nu} d(\theta, \theta') \right),$$

where $d(\theta, \theta') = \sqrt{(\theta - \theta')^\top L^{-2}(\theta - \theta')}$ is a scaled Euclidean distance with $L = \text{diag}(l_1, \dots, l_d)$, ζ is a scaling factor for the output variance, and Γ and B_ν are the Gamma and modified Bessel functions, respectively. The rate parameter $\nu \in (0, \infty)$ controls the smoothness of the prior and is a valid representation as long as f^i has continuous derivatives of any order less than ν (i.e., is at least $\lceil \nu - 1 \rceil$ times differentiable). Interested readers are referred to [27] for details on the different hyperparameters.

Traditional BO methods take advantage of the statistical information embedded in the GP approximations to intelligently explore the search space Θ by defining a corresponding acquisition function. The value of the acquisition function should provide a good measure of the potential benefit of querying the performance function f^0 at any particular $\theta \in \Theta$. Several acquisition functions have been proposed in the literature, with one of the most popular being the so-called expected improvement (EI) function, given by

$$\text{EI}_n(\theta) = \mathbb{E}_n[(f^0(\theta) - (f_n^0)^*)^+], \quad (6)$$

where $a^+ := \max(a, 0)$ and $(f_n^0)^*$ is the incumbent solution that corresponds to the best objective value observed so far. By $\mathbb{E}_n$, we refer to the expectation with respect to the posterior distribution at iteration n . The following closed-form expression for EI has been derived for GP approximators [28]:

$$\text{EI}_n(\theta) = \rho(\mu_n^0(\theta) - (f_n^0)^*, \sigma_n^0(\theta)), \quad (7)$$

where $\sigma_n^0(\theta) = \sqrt{k_n^0(\theta, \theta)}$ is the posterior predicted standard deviation for the performance function, ρ is given by

$$\rho(y, s) = \begin{cases} y\Phi(y/s) + s\phi(y/s), & s > 0, \\ \max(y, 0), & s = 0, \end{cases} \quad (8)$$

and Φ and ϕ , respectively, denote the cumulative distribution function (CDF) and probability density function (PDF) of a standard normal distribution, i.e.,

$$\Phi(x) = \frac{1}{2} \left[1 + \text{erf}\left(\frac{x}{\sqrt{2}}\right) \right], \quad \phi(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right),$$

and $\text{erf}(\cdot)$ is the error function.

3.2 Constrained Bayesian Optimization

To account for the black-box safety constraints, we need to ensure that the sequence of candidates $\{\theta_1, \dots, \theta_N\}$ (for any choice of N) will not violate the safety constraints. We will denote the safety set by

$$\mathcal{S} = \{\theta \in \Theta : f^i(\theta) \geq 0, \forall i \in \mathbb{N}_1^m\}, \quad (9)$$

where $\mathbb{N}_1^m := \{1, 2, \dots, m\}$. Therefore, our algorithm must ensure that $\theta_1, \dots, \theta_N \in \mathcal{S}$; however, since we do not have exact knowledge of the constraint functions, we cannot ascertain $\mathcal{S}$ *a priori*. However, we can use the GP models to construct an approximation of this safety set that is guaranteed to be an inner approximation (with high probability) via

$$\hat{\mathcal{S}}_n = \{\theta \in \Theta : \mu_n^i(\theta) - \sqrt{\beta_{n+1}} \sigma_n^i(\theta) \geq 0, \forall i \in \mathbb{N}_1^m\}, \quad (10)$$

where $\beta_{n+1} \geq 0$ is a weighting parameter on the variance term. Note that

$$\ell_n^i(\theta) := \mu_n^i(\theta) - \sqrt{\beta_{n+1}} \sigma_n^i(\theta)$$

represents a lower confidence bound for the unknown function f^i that, for properly chosen β_{n+1} , must be a lower bound on the true function with high probability (under some assumptions). If this holds, then $\hat{\mathcal{S}}_n \subseteq \mathcal{S}$ such that $\theta_{n+1} \in \hat{\mathcal{S}}_n \Rightarrow \theta_{n+1} \in \mathcal{S}$.

Classical constraint-enforcing BO proposes scaling the acquisition function by the probability of feasibility [29]. For example, the expected improvement with constraints (EIC) acquisition function is given by

$$\text{EIC}_n(\theta) = \text{EI}_n(\theta) \prod_{i=1}^m \Phi\left(\frac{\mu_n^i(\theta)}{\sigma_n^i(\theta)}\right), \quad (11)$$

where $\mathbb{P}_n\{f^i(\theta) \geq 0\} = \Phi(\mu_n^i(\theta)/\sigma_n^i(\theta))$ is the probability that the i^{th} constraint is satisfied according to the posterior distribution at iteration n . Sampling by maximizing EIC, while discouraging constraint violation, can actually result in frequent constraint violations in practice due to its soft penalization.

3.3 Proposed Safe LSR-BO Algorithm

To ensure safety, therefore, we wish to avoid sampling in regions that are not deemed (at least with satisfactorily high probability) to be safe. This can be done by sequentially solving the following constrained optimization problem

$$\theta_{n+1} \in \arg\max_{\theta} \{\text{EI}_n(\theta) : \theta \in \hat{\mathcal{S}}_n\}. \quad (12)$$

Note that, in the presence of constraints, the incumbent $(f_n^0)^*$ in (6) should now be replaced by the best *feasible* observation, denoted by f_c^* . Since EI_n and the safety constraints (10) are inexpensive to query (and if properly designed, gradient information is available from the GP kernels), this problem can be solved using well-known nonlinear optimization techniques [30]. However, one can use a simplification [24] to avoid solving this nonlinearly constrained optimization problem, by replacing (12) with a log-barrier penalty

$$\theta_{n+1} \in \arg\max_{\theta \in \Theta} \{\text{EI}_n(\theta) - \tau \sum_{i=1}^m \Gamma_{\ell_n^i}(\theta)\}, \quad (13)$$

where $\Gamma_{\ell_n^i}(\theta) = -\ln(\ell_n^i(\theta))$ is the logarithmic barrier function [30] applied to a constraint $\ell_n^i(\theta) \geq 0$ and $\tau > 0$ is a tunable parameter that ensures the barrier term converges to the exact indicator penalty function in the limit $\tau \rightarrow 0$.

Our proposed algorithm, which we refer to as safe LSR-BO, leverages this EI acquisition function with log-barrier in two ways. First, it performs a local optimization over the

Algorithm 1 Safe LSR-BO

Require: Domain Θ , initial dataset $\mathcal{D}_0 = \{x_0, \{f^i(x_0)\}_{i=0}^m\}$, $m+1$ prior GP models (4), confidence bound parameters $\{\beta_{n+1}^i\}_{n \geq 0}$, barrier parameter $\tau > 0$, and switching parameter $\gamma \geq 0$.

- 1: **for** $n = 0, 1, \dots$ **do**
 - 2: Construct GP posterior for $\{f^i\}_{i=0}^m$ given $\mathcal{D}_n$ via (5).
 - 3: $\theta_{n+1}^{\text{local}} \leftarrow \arg\max_{\theta \in \mathcal{B}_{\delta}(\theta_n)} \{\text{EI}_n(\theta) - \tau \sum_{i=1}^m \Gamma_{\ell_n^i}(\theta)\}$
 - 4: $\theta_{n+1}^{\text{global}} \leftarrow \arg\max_{\theta \in \Theta} \{\text{EI}_n(\theta) - \tau \sum_{i=1}^m \Gamma_{\ell_n^i}(\theta)\}$
 - 5: Obtain θ_{n+1}^* using safe switching rule (15)
 - 6: Query at θ_{n+1}^* and observe cost $y_{n+1} = \hat{f}(\theta_{n+1}^*)$
 - 7: Update dataset $\mathcal{D}_{n+1} \leftarrow \mathcal{D}_n \cup \{\theta_{n+1}^*, \{f^i(\theta_{n+1}^*)\}_{i=0}^m\}$
 - 8: **end for**
-

LSR constraints (3) to find the best point $\theta_{n+1}^{\text{local}}$ reachable in one step, that is:

$$\theta_{n+1}^{\text{local}} \in \arg\max_{\theta \in \mathcal{B}_{\delta}(\theta_n)} \{\text{EI}_n(\theta) - \tau \sum_{i=1}^m \Gamma_{\ell_n^i}(\theta)\}. \quad (14)$$

While $\theta_{n+1}^{\text{local}}$ must satisfy LSR constraints, it is possible that no nearby point has a reasonable chance of improving upon the current incumbent – this will happen whenever the region $\mathcal{B}_{\delta}(\theta_n)$ and its neighborhood has been sufficiently well-explored. When such situations arise, a reasonable alternative is to find the globally best point by solving (13) for $\theta_{n+1}^{\text{global}} \leftarrow \theta_{n+1}^*$. Clearly, we cannot always run an experiment with $\theta_{n+1}^{\text{global}}$ because it might violate the LSR constraint (3). The most straightforward way to project this point onto

$$\hat{\mathcal{F}}_n(\delta) = \mathcal{B}_{\delta}(\theta_n) \cap \hat{\mathcal{S}}_n,$$

which represents our current approximation of the feasible set (feasible implies that it satisfies both safety and LSR constraints). Using such a projection approach at every iteration, however, would result in us potentially missing out on local improvements that may be available as we move throughout the feasible space. The key idea in safe LSR-BO is to automatically tradeoff between these “local” and “global” steps in (14) and (13) using the following condition

$$\theta_{n+1}^* = \begin{cases} \theta_{n+1}^{\text{local}}, & \text{EI}_n(\theta_{n+1}^{\text{local}}) \geq \gamma, \\ \text{Proj}_{\hat{\mathcal{F}}_n(\delta)}(\theta_{n+1}^{\text{global}}), & \text{otherwise,} \end{cases} \quad (15)$$

where γ is a user-defined tuning parameter; increasing γ restricts the degree of local search. We can see that (15) reduces to greedy local and a global search as $\gamma \rightarrow 0$ and $\gamma \rightarrow \infty$, respectively. We found that a reasonably small value of $\gamma = 0.01$ appears to work well in practice. Pseudocode for the safe LSR-BO algorithm is summarized in Algorithm 1.

Safe LSR-BO is extremely simple to implement and has computational and memory requirements on the same order as traditional (safe) BO methods. Furthermore, as we show in the next section, we can guarantee it will satisfy safety

constraints for any choice of γ . We are also able to guarantee convergence to the global solution, i.e., $(f_n^0)^* \rightarrow f^0(\theta^*)$ in the absence of safety constraints.

3.4 Handling Noisy Observations in Safe LSR-BO

Suppose now that we do not directly measure $f^i(\theta_j)$ and instead measure noise-corrupted values $y_j^i = f^i(\theta_j) + \varepsilon_j^i$ for all $i = 0, \dots, m$ and $j = 1, \dots, n$ where ε_j^i represents the observation noise. The proposed safe LSR-BO method described in the previous section does not directly handle noise since we no longer can identify the feasible incumbent f_c^* in the presence of noise. However, we can take advantage of recent methods that extend EI to the noisy observation case. These methods rely on the fact that GP models can be easily applied to noisy measurements whenever $\varepsilon_j^i \sim \mathcal{N}(0, (\lambda_j^i)^2)$ is normally distributed where λ_j^i denotes the standard deviation of the j^{th} measurement of the i^{th} unknown function. The structure of the posterior GP equations remain the same as (5), with the vector of true function values $\mathbf{f}_n^i$ being replaced by the vector of noisy observations $\mathbf{y}_n^i$ and $\mathbf{K}_n^i$ being updated to include the noise characteristic $\mathbf{K}_n^i + \text{diag}((\lambda_1^i)^2, \dots, (\lambda_n^i)^2)$ (see, e.g., [31, Chapter 2] for details). We can use the GP posterior to effectively “filter” the noise from the observation when defining the incumbent.

Two strategies for reducing the impact of noise, both of which can be straightforwardly incorporated into the proposed safe LSR-BO algorithm, are briefly discussed next. First, we can use the idea of a “plug in” estimate to define f_c^* . Following [32], the incumbent can be defined as

$$f_c^* = \max_{\theta \in \Theta} \mathbb{E}_n \{f^0(\theta)\}, \quad (16)$$

$$\text{subject to: } \mathbb{P}_n \{f^i(\theta) \geq 0, \forall i \in \{1, \dots, m\}\} \geq 1 - \delta,$$

where $\mathbb{E}_n$ and $\mathbb{P}_n$ denote the conditional expectation and probability given all data up until iteration n , respectively, and δ is the maximum allowed violation probability. The probabilistic functions in (16) can be analytically expressed in terms of the GP posterior mean and variance functions such that f_c^* can be found with minimal computational effort. As discussed in [33], this heuristic for setting f_c^* can be highly exploitative in the high-noise case since it results in high EI values for replicating prior experiments until their probability of feasibility is driven above $1 - \delta$. An alternative approach to the plug-in method is the noisy EI (NEI) acquisition function defined as follows [33]:

$$\text{NEI}_n(\theta) = \int_{\mathbf{f}_n^0, \dots, \mathbf{f}_n^m} \text{EIC}_n(\theta \mid \mathbf{f}_n^0, \dots, \mathbf{f}_n^m) \prod_{i=0}^m p(\mathbf{f}_n^i \mid \mathcal{D}_n^0) d\mathbf{f}_n^i,$$

where $\text{EIC}_n(\theta \mid \mathbf{f}_n^0, \dots, \mathbf{f}_n^m)$ denotes the standard EIC acquisition function in (11) given noise-free observations $\{\mathbf{f}_n^i\}_{i=0}^m$. NEI is an intuitive extension of EI to the noisy case since one is simply averaging over realizations of the true function values at the sample locations drawn from the posterior GP. The

main challenge with NEI is that we can no longer derive an analytic expression for it; however, since we can easily sample from the posterior GP, NEI and its gradient with respect to θ can be efficiently estimated with quasi-Monte Carlo integration. Therefore, we suggest using NEI only in cases where the noise variance levels are expected to be quite high relatively to the true function values.

4 Theoretical Analysis of Safe LSR-BO

In this section, we establish two useful properties of the safe LSR-BO method. First, we show that, under a few technical assumptions, it guarantees safety constraints are satisfied at all iterations up to some user-specified probability. These safety constraints do place a fundamental limitation on performance since we may not be able to accurately certify a value $\theta \in \Theta$ as safe or unsafe unless we are willing to sample at potentially unsafe locations. Second, we establish a few convergence results for safe LSR-BO, which depend on the behavior of the estimated safe set $\hat{S}_n$.

4.1 Safe Learning Property

To establish that the proposed safe LSR-BO algorithm guarantees the black-box safety constraints are satisfied at every iteration, we need to make one major assumption that the GP models for the constraints are sufficiently well-calibrated, as described below.

Assumption 1 (Well-calibrated GPs). *The GP models for the unknown constraint functions $\{f^i\}_{i=1}^m$ satisfy the inequality below for all $\theta \in \Theta$, $n \geq 0$, and $i = 1, \dots, m$*

$$|f^i(\theta) - \mu_n^i(\theta)| \leq \sqrt{\beta_{n+1}^i} \sigma_n^i(\theta), \quad (17)$$

with probability at least $1 - \delta$ for some $\delta \in (0, 1)$. $\triangleleft$

As shown in [34, Theorem 2], one can select the confidence bound parameters $\{\beta_{n+1}^i\}_{n \geq 0}$ in a way that ensures Assumption 1 is satisfied as long as all f^i have a bounded reproducing kernel Hilbert space (RKHS) norm. This requirement is satisfied by any function that meets some reasonable continuity and smoothness requirements. If these functions were arbitrarily discontinuous, then two neighboring values could be completely independent/uncorrelated and there would be no way to guarantee safety in our problem setting (assumption is necessary to make progress). We are now able to formally state the safety guarantees conferred by (15).

Theorem 1 (Safe Learning Guarantees). *Let Assumption 1 hold for some $\delta \in (0, 1)$, the safety set S be non-empty, and there exists at least one known safe starting point $\theta_0 \in S$. Then, the sequence of query points $\{\theta_n\}_{n \geq 1}$ generated by (15) satisfies*

$$\mathbb{P} \{f^i(\theta_n) \geq 0, \forall i \in \mathbb{N}_1^m, \forall n \geq 1\} \geq 1 - \delta. \quad (18)$$

Proof. From Assumption 1, we must have

$$f^i(\theta) \geq \mu_n^i(\theta) - \sqrt{\beta_{n+1}^i \sigma_n^i(\theta)}, \quad \forall \theta \in \Theta, \forall i \in \mathbb{N}_1^m, \forall n \geq 0,$$

holds with probability $\geq 1 - \delta$. Given access to at least one safe point $\theta_0 \in \mathcal{S}$, we know that the partially revealed safe region must be non-empty $\hat{\mathcal{S}}_n \neq \emptyset$ for all $n \geq 0$. From the definition of the log barrier term, we know that the solution to (14) must satisfy the following restricted constraints

$$\mu_n^i(\theta_{n+1}^{\text{local}}) - \sqrt{\beta_{n+1}^i \sigma_n^i(\theta_{n+1}^{\text{local}})} \geq 0, \quad \forall i \in \mathbb{N}_1^m.$$

A similar result holds for the projected point in (15) since the same constraints are enforced by the projected operation, which implies these constraints are also satisfied by θ_{n+1} . Combining these two inequalities proves the result. ■

Intuitively, Theorem 1 states that one can expect safety constraints to be satisfied as long as the trained GP models are able to capture the true function within its confidence bounds with high probability. It also involves a very natural parameter δ , which is the maximum allowable violation probability. As $\delta \rightarrow 0$, the confidence bounds will necessarily grow, leading to a reduction in the volume of $\hat{\mathcal{S}}_n$. In other words, excessive caution during the optimization process greatly limits the set of testable controller tuning parameters at each iteration.

Remark 1. The confidence bound parameters β_{n+1}^i can be freely set by the user in accordance with desired safety requirements. Larger values, by definition, will reduce the likelihood of violation and shrink $\hat{\mathcal{S}}_n$ to the set of previously sampled safe points as $\beta_{n+1}^i \rightarrow \infty$. The values specified in [34, Theorem 2] have been proven to be sufficiently large to guarantee Assumption 1, though smaller values have been found to work reasonably well in practice. It is quite common to select $\beta_{n+1}^i \in [4, 9]$, which ensures that 95% to 99.7% of the predictions for $f^i(\theta)$ lie above $\ell_n^i(\theta)$ as a result of the outcome having a Gaussian distribution. Alternative ways to select β_{n+1}^i have been studied in [35, 36], which is something we plan to investigate further in future work.

4.2 Convergence Under LSR Constraints

We analyzed the convergence properties of LSR-BO in the absence of constraints in our previous work [26]. We now extend these results to the safety constraint setting. We need one critical assumption on the covariance function k^0 of the performance function f^0 , described next.

Assumption 2 (NEB Kernel). The GP model of the performance function has a kernel k^0 that satisfies the no-empty-ball (NEB) property such that

$$\inf_{\substack{n \geq 0 \\ \theta_1, \dots, \theta_n \in \Theta \\ \|\theta_i - \theta_0\| \geq \epsilon, \forall i}} \sigma_n^0(\theta_0) > 0$$

for any sequence $\{\theta_n\}_{n \geq 1}$, any $\epsilon > 0$, and any $\theta_0 \in \Theta$. ◁

As shown in [37, Proposition 10], kernel functions from the Matérn class have the NEB property and such kernels can be used to reconstruct sufficiently smooth performance functions. We can now establish the following convergence result for safe LSR-BO.

Theorem 2. Let Assumptions 1 and 2 hold and assume that $\hat{\mathcal{S}} = \hat{\mathcal{S}}_{n^*+i}$ for all $i \geq 0$ (i.e., estimated safe set converges after some finite iteration n^*). Suppose for $\gamma > 0$, there exists a sequence $\{m_n\}_{n \geq 1}$ and $M < \infty$ such that the sequence generated by safe LSR-BO satisfies $\theta_{m_{n+1}} = \theta_{m_n+1}^{\text{global}}$ with $|m_{n+1} - m_n| \leq M$ for all $n \geq 0$. Then, the safe LSR-BO strategy generates a sequence $\{\theta_n\}_{n \geq 1}$ satisfying (18) and

$$f_c^* = \max_{j=1, \dots, n} \{f^0(\theta_j) : f^i(\theta_j) \geq 0\} \xrightarrow{a.s.} f^0(\theta_s^*), \quad (19)$$

as $n \rightarrow \infty$ and $\tau \rightarrow 0$, where $\theta_s^* \in \text{argmax}_{\theta \in \hat{\mathcal{S}}} f^0(\theta)$ is the optimal parameter in the converged safe set.

Proof. The safety result in terms of (18) being satisfied follows directly from Theorem 1. To prove (19), we first note that (13) converges to (12) as $\tau \rightarrow 0$. Further, by assumption, the safe set converges to a fixed set $\hat{\mathcal{S}}$ after some iteration n^* such that (12) can be interpreted as applying a standard EI strategy in the reduced space $\hat{\mathcal{S}} \subset \Theta$. Under the assumption on the sample points, we can always take a subsequence $\{\theta_{m_0}^{\text{global}}, \dots, \theta_{m_n}^{\text{global}}\}$ from the sequence generated by LSR-BO for any $n \geq 1$. These “global” samples can be at most a finite number of M steps apart in this sequence between any n and $n+1$. Since the safe set is fixed for any $n \geq n^*$, we know that global EI provides dense sampling (almost surely) in $\hat{\mathcal{S}}$ under the NEB assumption, as proved in [37, Theorem 6]. This is equivalent to the statement in (19). ■

The additional assumptions made in Theorem 2 deserve further elaboration. The first key assumption is that the estimated safety set converges after some finite iteration. We do not attempt to formally prove this here; however, we note that this can always be achieved in practice by no longer performing posterior updates after a certain period of time. In other words, reuse the functions $\{\ell_{n^*}^i(\theta)\}$ for all future $n \geq n^*$ iterations, which will also reduce computational cost since the GP predictions scale with the number of data points. The choice of n^* is flexible, though it is recommended to wait until enough data has been collected such that most points on the boundary of $\hat{\mathcal{S}}_n$ has reasonably low variance. The second key assumption is about the existence of a sequence $\{m_n\}_{n \geq 0}$ that makes LSR-BO match the global EI policy. Although formally proving this is difficult, we can simply modify LSR-BO to make it satisfy this requirement by construction by employing a terminal constraint $\theta_{n+M} = \theta_{n+1}^{\text{global}}$ for all $n \in \{0, M, 2M, \dots\}$. The choice of M needs to be at least as large as the minimum number of steps required to move between any two points in Θ , which is finite for any $\delta > 0$.

A final implicit assumption is that the hyperparameters of the covariance functions (such as length- and output-scales) are known in advance. Since we consider black-box functions in this work, these values are often unknown and must be estimated from observed data. Typically, these estimates are updated at every iteration using, e.g., maximum likelihood estimation; however, as shown in [38], the global EI policy does not automatically converge in this case, as the GP model may underestimate the predicted variance for certain observations. We can use the same modifications as proposed in [38, Definition 3] to show that LSR-BO converges in this more general case.

Note that Theorem 2 only guarantees convergence to the true global maximum if $\theta^* \in \mathcal{S}$, which is not necessarily satisfied. One way to ensure this holds is to guarantee $\hat{\mathcal{S}}_n \rightarrow \mathcal{S}$ as $n \rightarrow \infty$; however, this is not true in general. Imagine the case that Θ is composed of discrete elements and there is one safe point that is far away from any other neighboring safe point – this point will never be certified as safe unless queried (since it is too far away from any neighbor to ensure its lower confidence bound satisfies constraints). However, since we can only query certifiably safe points, this point will always be excluded from our search process. If this point happened to be the global maximizer, then our method would be unable to converge to the true solution. Recent work has shown that one can guarantee convergence to the global maximizer in the presence of constraints by *relaxing* constraints [39, 40] but these methods do not guarantee any form of safety during the search process. Convergence of safe BO remains an open research challenge and is something we plan to study more in our future work.

5 Benchmarking and Case Study

In this section, we present two examples that illustrate the effectiveness of our proposed LSR-BO method. The method was implemented using BOTorch [41] with default settings. In particular, GP priors with $\mu^i(\theta) = 0$ and $k^i(\theta, \theta')$ from the Matérn class with $\nu = 2.5$ for all $i = 0, \dots, m$. The safe EI acquisition function in (13) was maximized L-BFGS-B algorithm [42] wherein the best 10 points from an initial set of 1000 samples are used as the starting points. For each example, we give a description of the problem and show statistical comparisons with competing alternative methods.

5.1 2-D Benchmark Example

The unknown performance function $f^0(\theta)$ is given by a modified version of the Branin function from [43], which is a benchmark problem for global optimization algorithms due to its non-convex nature with multiple local optima

$$f^0(\theta) = -\sum_{i=1}^4 f_i^0(\theta), \quad (20)$$

where

$$\begin{aligned} f_1^0(\theta) &= a(\theta_2 - b\theta_1^2 + c\theta_1 - r)^2, \\ f_2^0(\theta) &= s(1-t)\cos(\theta_1) + s, \\ f_3^0(\theta) &= 5e^{-5((\theta_1^2+3.14)^2+(\theta_2^2-12.27)^2)}, \\ f_4^0(\theta) &= 5e^{-5((\theta_1^2-3.14)^2+(\theta_2^2-2.275)^2)}, \end{aligned}$$

with constant values given by $a = 1$, $b = 5.1/(4\pi^2)$, $c = 5/\pi$, $r = 6$, $s = 10$, and $t = 1/(8\pi)$.

The compact domain is given by $\Theta = [-5, 10] \times [0, 15]$. We also require LSR constraints to be satisfied to mimic the requirement of safe exploration, which are given by

$$|\theta_{1,n+1} - \theta_{1,n}| \leq 0.5, \quad |\theta_{2,n+1} - \theta_{2,n}| \leq 1.5, \quad (21)$$

that must hold for all $n \geq 0$. These constraints force the next sample to be within a hyperrectangle of the current sample, which can be straightforwardly mapped to a hypercube of the form (3) with $p = \infty$. Black-box safety constraints will be considered in the next section.

We select *simple regret* (SR) as our key performance index, which is defined as follows

$$\text{SR}_n(\mathcal{D}_0^0) = f^0(\theta^*) - (f_n^0)^* = f^0(\theta^*) - \max_{(\theta, y) \in \mathcal{D}_n^0} y, \quad (22)$$

and depends on the initial performance data $\mathcal{D}_0^0$ and the current iteration n . SR simply measures the distance between the global solution and the current best sample. To show that LSR-BO performs well across a wide array of starting points, we randomly generate 50 initial datasets $\mathcal{D}_0^0$ composed of 10 samples generated in quasi-uniform random fashion inside of Θ . The starting value θ_0 is always selected to the point in $\mathcal{D}_0^0$ that maximizes the performance function. We use these replicates to estimate statistical properties of the sequences generated by LSR-BO as well as several competing alternatives, described below.

Shortest-Path: The global EI problem is solved to generate the next desired sample. If the global point is outside of the LSR, we sample the sequence of points along the shortest path between the current and desired global point.

Projection: We sample by projecting the global EI point onto the LSR constraints, i.e., $\theta_{k+1} = \text{Proj}_{\mathcal{B}_\delta(\theta_k)}(\theta_{k+1}^{\text{global}})$. We can think of this as a special case of the LSR-BO sampling method (15), when $\gamma \rightarrow \infty$.

Random: The procedure is the same as the shortest-path algorithm, except that the desired point is selected uniformly at random. This approach is thus a simple modification to random search to ensure satisfaction of LSR constraints.

TuRBO: The TuRBO method is a popular trust region-based BO algorithm presented in [44]. The size of the trust region is automatically updated in TuRBO based on the performance of the objective function value obtained at each sample. To ensure TuRBO satisfies LSR constraints, we only

Table 1. Average cost per iteration (in seconds) for all methods.

LSR	Projection	Shortest-path	TuRBO	Random
0.54	0.55	0.12	0.81	0.08

allow the trust region size to grow to a maximum radius value of δ . Furthermore, although TuRBO can run several trust regions in parallel, we select a single trust region to ensure a fair efficiency comparison with the other methods. While we modify the core TuRBO algorithm to make a fair baseline, note that there are two key differences with TuRBO which makes our LSR-BO algorithm quite different both in concept and in implementation. First, TuRBO uses the trust region to limit where the model is constructed (inherently local) whereas LSR-BO uses a global GP to understand the full landscape. Second, TuRBO adaptively updates the size of the trust region depending on the quality of the step, so it does not have a mechanism to limit growth and satisfy the safety constraints.

The statistical results for $SR_n(\mathcal{D}_0^0)$ for all methods up to $n = 80$ iterations are shown in Figure 1. We see that LSR-BO clearly outperforms all considered methods, achieving up to 1-4 orders of magnitude improvement in the median and 90% confidence interval values compared to the baseline methods. In particular, TuRBO performs poorly for two reasons: (1) It is a local method by design and we only use a single trust region to keep its implementation closer to LSR-BO. This means that it loses the globalization property that made it perform better in the original paper (notice that this requires substantially more samples overall and would not be implementable on a physical system). (2) Since TuRBO does not limit growth of trust region, we needed to artificially limit it to a maximum value to satisfy LSR constraints. This likely introduces performance losses since it only has a local view of the function. Furthermore, we see that LSR-BO consistently makes improvement in the early stages (indicative of a quickly finding a nearby local solution) and then expands outward toward the global solution. Note that the average computational cost per iteration for all considered optimization methods are also shown in Table 1. We see that the cost of LSR-BO is less than one second per iteration, implying the improved performance does come at the cost of increased planning time relative to alternative methods.

We also study the impact of the hyperparameter γ on the performance of LSR-BO in Figure 2 wherein we show a violin plot for the distribution of final simple regret values obtained for $\gamma \in \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1\}$. It is interesting to note that for a sufficiently small γ value (below 0.1), we see similar overall performance after 80 iterations. There is a slight increase in the worst-case sample of the simple regret value for very small choices of γ but this is relatively small (as we can also observe in Figure 1). As such, it seems like one should preferentially set smaller γ values to emphasize local exploitation, though very small values definitely have the tendency to result in performance losses. The degree of those losses will depend heavily on the suboptimality of the nearby local optimum compared to the global

./figures/converg_box.png

Figure 1. Comparison of simple regret statistics (median and 90% confidence interval) versus number of iterations for proposed (LSR-BO), shortest-path, projection, random, and TuRBO algorithms.

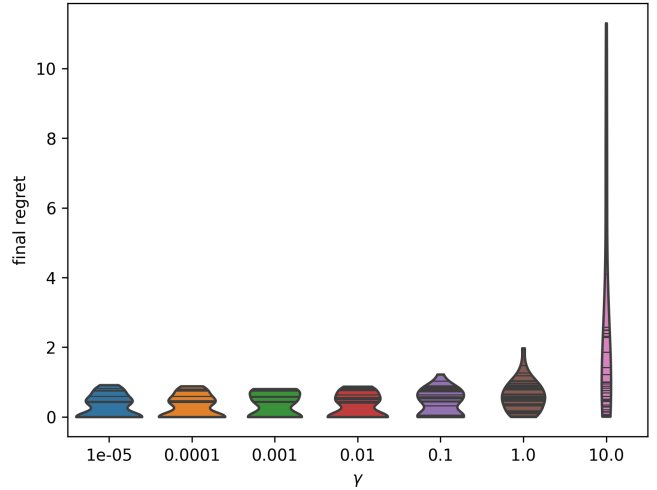


Figure 2. Violin plot for the final simple regret values obtained with LSR-BO under different γ values.

optimum, so future work is needed to more generally understand the impact of γ on performance. The development of an adaptive procedure to set γ based on the expected quality of the local solution may be an interesting direction to pursue.

In addition to the LSR constraints, we additionally incorporate a black-box safety constraint of the form

$$f^1(\theta) = \theta_1 - \theta_2 - \sin(\theta_2) + (\theta_1/4)^2. \quad (23)$$

To highlight the importance of the safe BO formulation, we compare the performance of the safe formulation in (15) to the EIC formulation wherein we replace the barrier-based ac-

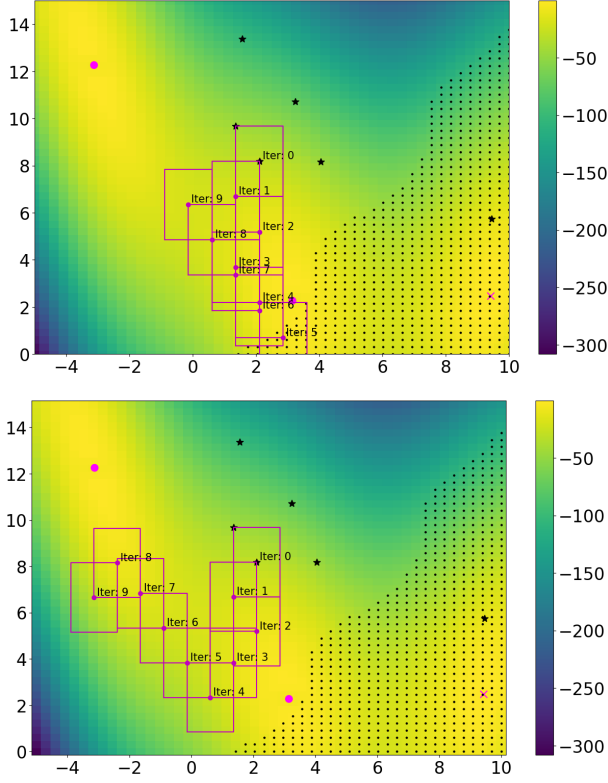


Figure 3. Comparison of black-box safety constraint handling capabilities of the EIC (top) and the proposed safe LSR-BO (bottom) acquisition functions. The black stars denote the initial data, the magenta boxes denote the LSR constraints, and the large magenta dots denote optimal solutions that (locally) maximize the performance function subject to safety constraints.

quisition function with the EIC function in (11). The resulting sampling trajectory generated by both of these methods are shown in Figure 3. We clearly observe that EIC ends up violating the safety constraint whereas the safe LSR-BO method satisfies constraints at all iterations. We also observe that the proposed method is able to more effectively identify a path of high-performance values. As such, the method is able to practically enable safe learning without significant performance losses in this case, which provides confirmation of the theoretical results established in Section 4.

We also demonstrate the ability of LSR-BO to accommodate noisy observations by using the plug-in incumbent strategy described in Section 3.4. The statistical results for $SR_n(\mathcal{D}_0^0)$ (now with respect to random initial data and measurement noise) for all methods up to $n = 80$ iterations are shown in Figure 4. We see that LSR-BO continues to outperform all other considered methods, reaching a simple regret value of the same order as the noise standard deviation within around 40 evaluations.

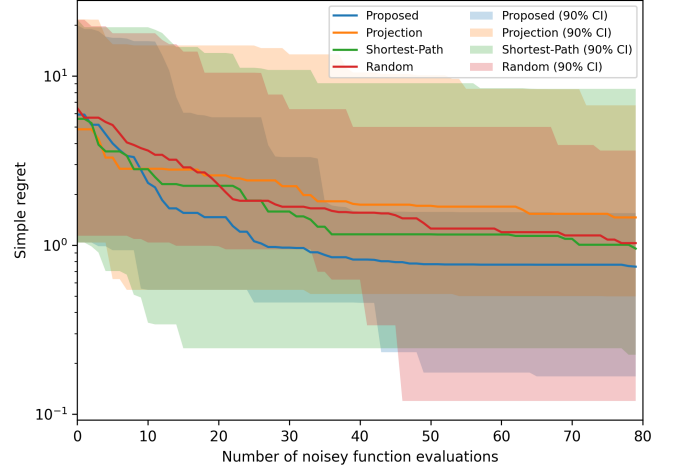


Figure 4. Comparison of simple regret statistics (median and 90% confidence interval) versus number of iterations for proposed (LSR-BO), shortest-path, pprojection, and TuRBO algorithms in the case of noisy observations with noise variance $\lambda^2 = 1$.

5.2 Case Study: Energy Self-Optimization for Industrial VCCs

5.2.1 With LSR constraints: LSR-BO

In this paper, the main purpose of the proposed LSR-BO algorithm is to tackle the real-world problem of tuning setpoints of a vapor-compression heat pump to minimize the operating power consumption while enforcing implementation and safety constraints.

A block diagram of the heat pump of interest is shown in Fig. 5A, which consists of a compressor, a condenser, an expansion valve, and an evaporator that exchanges heat between an indoor occupied setting and the ambient environment. We can cast this problem in the form of (2) by defining the performance function $f^0(\theta) = -P_\infty(\theta)$ as the negative of the steady-state power consumption P_∞ that is a function of the VCC with closed-loop references set to θ . In the VCC, we first close a feedback loop from compressor frequency to room temperature, which leaves us with three tunable setpoints θ representing the electrical expansion valve (EEV) position, the indoor fan speed (IFS), and the outdoor fan speed (OFS). By assigning fixed setpoint values θ , we wait for an adequate amount of time until the power signal resides within a 95% settling zone and use that to represent $J(\theta)$: from domain experience, this takes around 10 min after a setpoint change. The setpoints search space is defined by the following ranges: $EEV \in [200, 300]$ counts, $IFS \in [200, 500]$ rpm, and $OFS \in [500, 1000]$ rpm.

As mentioned before, recklessly changing these actuator setpoints can result in wear and tear of the actuators, and lead to undesired transients in the refrigerant flow and thermal dynamics. To avoid these harmful operating modes, we restrict the setpoint changes via the LSR constraints (3) and restrict the valve positions to change at most ± 5 counts, IFS at most ± 10 rpm, and OFS at most ± 25 rpm in consecutive LSR-BO iterations. An illustrative schematic of the proposed data-driven optimization approach for the heat pump system

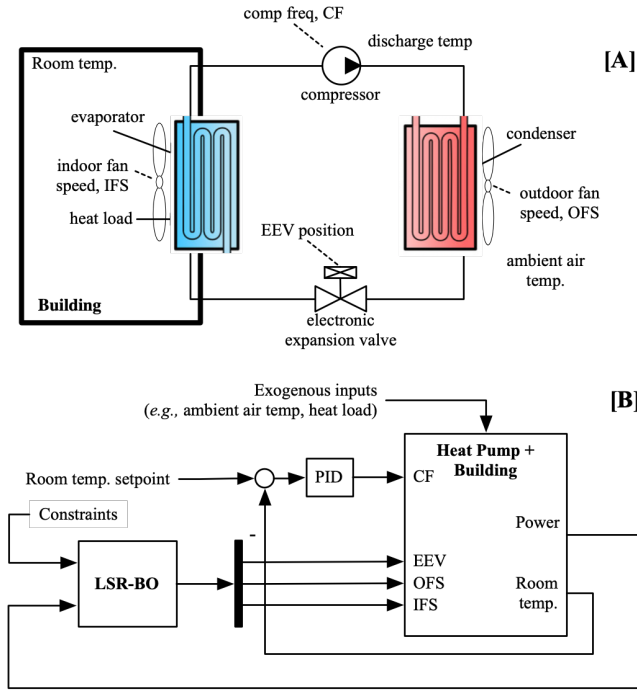


Figure 5. (A) Schematic of industrial heat pump system. (B) Illustration of the proposed LSR-BO method for minimizing power consumption through manipulation of setpoints.

is shown in Fig. 5B.

To validate our method, we use a high-fidelity dynamic model of a prototype vapor-compression system (VCS)¹ written in the *Modelica* language [45] to collect power consumption data and optimize the set-points on-the-fly. For more details about the model, we refer the reader to [14]. The simulation model was first developed in the *Dymola* [46] environment and then exported as a functional mockup unit (FMU) [47] for connecting with Python. The current version of the model comprises over 12000 differential equations and software modules that are not differentiable, motivating our use of zeroth-order/gradient-free optimization methods. We use the same GP model and optimization settings as discussed in Example 1.

We again compare the LSR-BO method to the shortest-path and projection methods defined previously. Since the true global optimum is unknown in this problem, we directly compare the power consumption values produced over 40 allowed high-fidelity function evaluations. Similar to Example 1, we initially populate $\mathcal{D}_0$ with high-fidelity evaluations at 10 domain-informed samples in Θ , and select θ_0 as the point within this set that produces the largest $J(\theta)$ (minimal power consumption). We again repeat all algorithms 50 times to estimate statistical properties of the generated sample trajectories. The resulting minimal power consumption profiles, i.e., $\min_{(\theta), P_{\infty}(\theta) \in \mathcal{D}_k} P_{\infty}(\theta)$ for the LSR-BO, shortest-path, and projection algorithms are shown in Fig. 6. We see

that that the proposed method achieves tighter confidence intervals, implying LSR-BO can reliably find setpoint values that lead to lower power consumption in a shorter amount of time. The complete dynamic simulation profiles for the VCS for the median LSR-BO run are also shown in Fig. 7. Note that the grey shading in the figure denotes the ‘offline’ experiments on the heat pump system: they are actually done online, which is why they appear in the trace, but with manual safety configurations running to avoid deleterious behavior. This is why there are LSR constraint violations in the grey shaded areas, and after that, the LSR constraints are consistently satisfied. Furthermore, we see a clear reduction in the power consumption over time, without generating any intermediate harmful transient states. We also note that the sensitivity of the power varies among the actuators; because the power is more sensitive to the EEV position, the EEV tends to converge to a steady-state value as the power is minimized. In comparison, the lower sensitivity of the power to the fan speeds, and in particular OFS, causes these actuators to continue to vary after power has converged to a minimum value because these variations do not translate to a significant change in the power consumption.

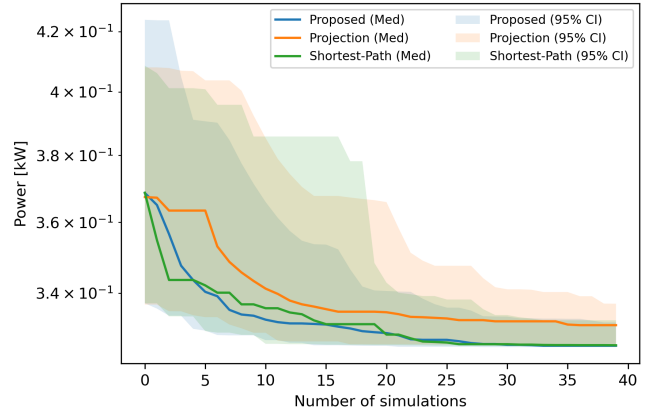


Figure 6. Comparison of minimum power statistics (median and 95% confidence interval) versus number of iterations for proposed (LSR-BO) method, shortest-path, and projection algorithms for the VCC.

5.2.2 With LSR and Safety Constraints: Safe LSR-BO

We also illustrate the self-optimization of VCCs with additional safety constraints. Two safety constraints are considered here: one is on the temperature of the refrigerant leaving the compressor, also referred to as the ‘discharge temperature’. Excessively high compressor discharge temperatures may break the composition of refrigerant oils and increase wear and tear. Furthermore, high discharge temperatures are often correlated with high fluid pressures, which can cause metal fatigue and compromise the integrity of the pressurized refrigerant pipes. The second constraint is on the evaporating temperature: this must be managed because of the effect

¹Note that while the behavior of this model have been validated against a real VCS, the numerical values and/or performance presented in this work is not representative of any product.

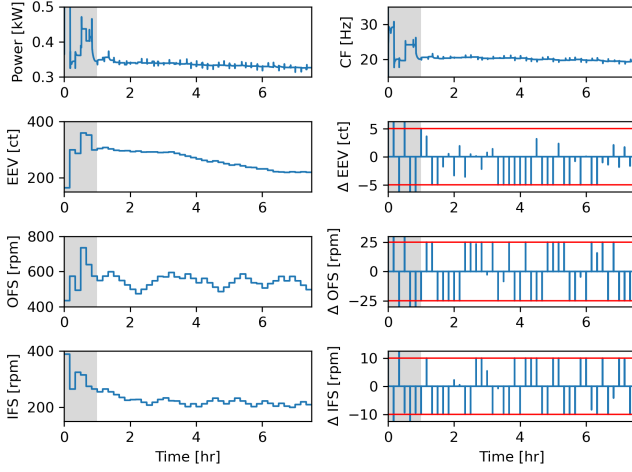


Figure 7. VCC power, compressor frequency (CF), EEV position, OFS, and IFS values over time for median LSR-BO run. ΔEEV , ΔOFS , and ΔIFS represent the change in setpoints at each iteration with the red lines denoting the LSR constraints. The grey shaded regions denote the offline experiments used to initialize the GP model.

that this temperature has on the accumulation of condensed water on the coil. As the evaporating temperature becomes smaller, significant amounts of condensation can accumulate on the heat exchanger, which will increase the pressure drop on the air traveling through the heat exchanger and potentially entrain water droplets into the air stream that could be expelled into the occupied space. In more extreme examples, evaporating temperatures below freezing will cause frost and ice to accumulate on the heat exchanger, reducing the cooling capacity of the cycle and potentially causing damage via the thermal expansion of the ice. Such constraints, among others, must clearly be managed effectively to ensure the safe and satisfactory operation of these variable-capacity vapor-compression systems. While managing the constraints mentioned above in the long run is critical, we also the flexibility that small violations over short periods of time have limited potential for harm. Concretely, we constrain the discharge temperature $T_{\text{dis}}(\theta) \leq 150^\circ\text{C}$ and $T_{\text{evap}}(\theta) \geq 23^\circ\text{C}$. In the format of (2), $f^1(\theta) := -150 + T_{\text{dis}}(\theta)$ and $f^2(\theta) := T_{\text{evap}}(\theta) - 23$. In order to make the algorithm more dexterous as it navigates complex feasible regions, we relax the LSR constraints to ± 10 counts, ± 20 rpm, and ± 50 rpm for the EEV, IFS, and OFS, respectively. To avoid constructing multiple GPs, we coalesce the two constraints into the single constraint: $\min\{f^1, f^2\} \geq 0$. Note that these constraint values are not indicative of the actual values tested for real-world deployment.

We compare the proposed safe LSR-BO with safe projection and safe shortest-path. We show the results of the power minimization performance in Fig. 8. As we have seen in the previous subsection, the proposed and shortest-path-based methods consistently outperform the projection approach, in terms of the median regret and confidence intervals obtained across 20 simulations (with unique initial setpoint values). An interesting thing to note from the figure

is that the proposed method converges faster and to a better minimal power value in comparison with the shortest-path variant as well, indicating that the local and global trade-off proposed in (15) are more important when the feasible region is more complex than having only an LSR constraint. This can be justified by understanding that the shortest-path approach is not flexible when the algorithm updates its feasible region as more constraint function data becomes available over iterations. Indeed, it is probably that the the shortest-path approach will move greedily in a direction that will not contain a feasible optimizer candidate; this greed is explicitly curtailed by our trade-off approach.

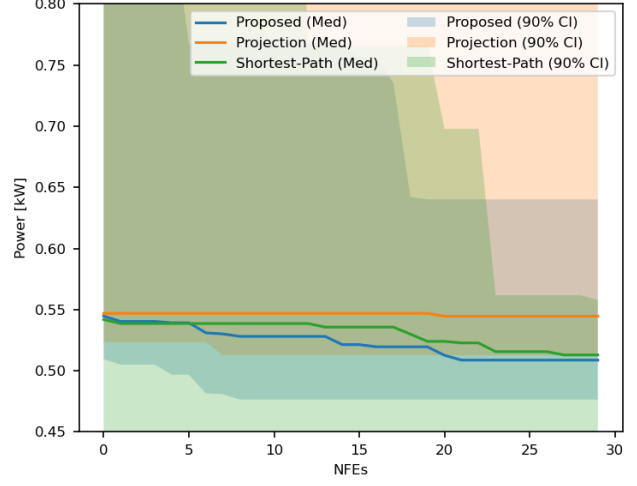


Figure 8. Comparison of minimum power statistics (median and 95% confidence interval) versus number of iterations for Safe LSR-BO, shortest-path, and projection algorithms for the VCC with temperature constraints.

An exemplar safe LSR-BO run is illustrated in Fig. 9. The LSR constraints are depicted by red horizontal lines, and the temperature constraints with magenta horizontal lines. The power plot shows that the safe LSR-BO exploits a direction in which the power exhibits lowering, until it achieves a feasible minimal power of 470 W around 3 hr. Then the algorithm begins to explore in search of a lower power value, and in its explorations, it slightly violates the evaporator constraint around the 4 hr mark when the evaporator temperature attains 22.98°C . The discharge temperature constraint is not violated after initialization; this is because the expansion valve is dilated for most of the run. The LSR constraints are never violated, although they are often grazed. Another interesting comparison is that with Fig. 7, where the minimum power was far below 470 W, since temperature violations were not explicitly considered there.

While we do not include those trajectories in the figure, we also make a comparison of the proposed approach against an LSR-BO of the form (15) but with the log-barrier removed and the EI acquisition replaced with a classical constrained EI acquisition function [29]; we refer to this as

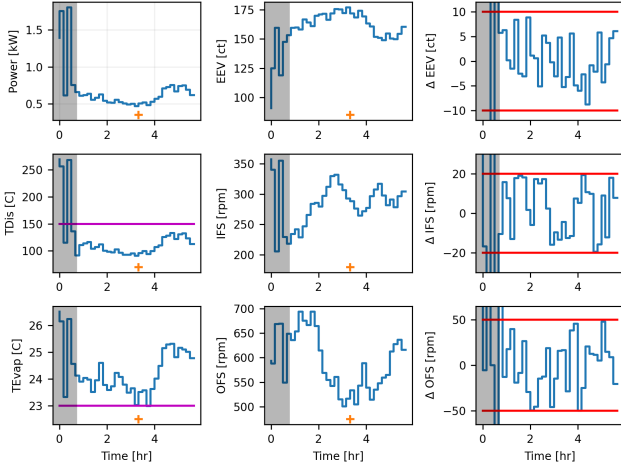


Figure 9. VCC power, compressor frequency (CF), EEV position, OFS, IFS, compressor discharge (TDis) temperature, and evaporating temperature (TEvap) values over time for median Safe LSR-BO run. Δ EEV, Δ OFS, and Δ IFS represent the change in setpoints at each iteration with the red lines denoting the LSR constraints, and magenta lines are safety constraints. The grey shaded regions denote the offline experiments used to initialize the GP model and the orange '+' indicates the optimal (and therefore, feasible) power value.

LSR-BO-EIC. Some interesting insights are obtained in this comparison. First, the minimal feasible power obtained by both methods are similar, but the median of safe LSR-BO is 470 W, whereas the median of LSR-BO-EIC is slightly higher at 478 W. Second, the LSR constraints are enforced by both methods, but the safety constraints are significantly worsened with LSR-BO-EIC. In fact, out of the 6 simulation hours, our safe LSR-BO violated an evaporating temperature constraint by 0.02°C for a sampling period, that is, 10 min. Alarming, the LSR-BO-EIC spends over 2 hr in constraint violated states, with violations as large as 50°C on the discharge temperature. This is primarily owing to the EIC function's philosophy of 'discouraging' safety constraint violations, rather than strongly penalizing them as we do with the log-barrier.

6 Conclusions

In this paper, we developed a novel Bayesian optimization framework that is capable of handling local search region constraints, which arise in a variety of practical engineering problems, especially for performance optimization of closed-loop dynamical systems such as vapor compression cycles. We also illustrate how LSR-BO can handle additional safety constraints via a log-barrier formulation. Our proposed method is extremely simple to implement and comes with theoretical guarantees of optimization and constraint-enforcement performance. There are several interesting directions for future work including improved understanding of the impact of the local/global switching parameter on performance, deriving explicit bounds on the convergence rate, and extensions to more complex observation noise models.

References

- [1] She, X., Cong, L., Nie, B., Leng, G., Peng, H., Chen, Y., Zhang, X., Wen, T., Yang, H., and Luo, Y., 2018. "Energy-efficient and-economic technologies for air conditioning with vapor compression refrigeration: A comprehensive review". *Applied Energy*, **232**, pp. 157–186.
- [2] Jain, N., Koeln, J. P., Sundaram, S., and Alleyne, A. G., 2014. "Partially decentralized control of large-scale variable-refrigerant-flow systems in buildings". *Journal of Process Control*, **24**(6), pp. 798–819.
- [3] Wallace, M., Das, B., Mhaskar, P., House, J., and Salsbury, T., 2012. "Offset-free model predictive control of a vapor compression cycle". *Journal of Process Control*, **22**(7), pp. 1374–1386.
- [4] Jäschke, J., and Skogestad, S., 2011. "Nco tracking and self-optimizing control in the context of real-time optimization". *Journal of Process Control*, **21**(10), pp. 1407–1416.
- [5] Garriga, J. L., and Soroush, M., 2010. "Model predictive control tuning methods: A review". *Industrial & Engineering Chemistry Research*, **49**(8), pp. 3505–3515.
- [6] Paulson, J. A., and Mesbah, A., 2020. "Data-driven scenario optimization for automated controller tuning with probabilistic performance guarantees". *IEEE Control Systems Letters*, **5**(4), pp. 1477–1482.
- [7] Paulson, J. A., Shao, K., and Mesbah, A., 2021. "Probabilistically robust Bayesian optimization for data-driven design of arbitrary controllers with Gaussian process emulators". In *Proceedings of the Conference on Decision and Control*, IEEE, pp. 3633–3639.
- [8] Xu, W., Jones, C. N., Svetozarevic, B., Laughman, C. R., and Chakrabarty, A., 2022. "VABO: Violation-aware Bayesian optimization for closed-loop control performance optimization with unmodeled constraints". In *Proc. American Control Conference (ACC)*, IEEE, pp. 5288–5293.
- [9] Rios, L. M., and Sahinidis, N. V., 2013. "Derivative-free optimization: a review of algorithms and comparison of software implementations". *Journal of Global Optimization*, **56**(3), pp. 1247–1293.
- [10] Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and De Freitas, N., 2015. "Taking the human out of the loop: A review of Bayesian optimization". *Proceedings of the IEEE*, **104**(1), pp. 148–175.
- [11] Frazier, P. I., 2018. "A tutorial on Bayesian optimization". *arXiv preprint arXiv:1807.02811*.
- [12] Baheri, A., and Vermillion, C., 2020. "Waypoint optimization using Bayesian optimization: A case study in airborne wind energy systems". In *Proceedings of the American Control Conference*, IEEE, pp. 5102–5017.
- [13] Pal, A., Zhu, L., Wang, Y., and Zhu, G. G., 2020. "Multi-objective stochastic Bayesian optimization for iterative engine calibration". In *Proceedings of the American Control Conference*, IEEE, pp. 4893–4898.
- [14] Chakrabarty, A., Danielson, C., Bortoff, S. A., and Laughman, C. R., 2021. "Accelerating

- self-optimization control of refrigerant cycles with Bayesian optimization and adaptive moment estimation”. *Applied Thermal Engineering*, **197**, p. 117335.
- [15] Chakrabarty, A., Burns, D. J., Guay, M., and Laughman, C. R., 2022. “Extremum seeking controller tuning for heat pump optimization using failure-robust bayesian optimization”. *Journal of Process Control*, **120**, pp. 86–96.
 - [16] Duivenvoorden, R. R., Berkenkamp, F., Carion, N., Krause, A., and Schoellig, A. P., 2017. “Constrained Bayesian optimization with particle swarms for safe adaptive controller tuning”. *IFAC-PapersOnLine*, **50**(1), pp. 11800–11807.
 - [17] Khosravi, M., Eichler, A., Schmid, N., Smith, R. S., and Heer, P., 2019. “Controller tuning by Bayesian optimization an application to a heat pump”. In *Proceedings of the European Control Conference*, IEEE, pp. 1467–1472.
 - [18] Lederer, A., Capone, A., and Hirche, S., 2020. “Parameter optimization for learning-based control of control-affine systems”. In *Learning for Dynamics and Control*, PMLR, pp. 465–475.
 - [19] Khosravi, M., Behrunani, V. N., Myszkowski, P., Smith, R. S., Rupenyan, A., and Lygeros, J., 2021. “Performance-driven cascade controller tuning with Bayesian optimization”. *IEEE Transactions on Industrial Electronics*, **69**(1), pp. 1032–1042.
 - [20] Piga, D., Forgone, M., Formentin, S., and Bemporad, A., 2019. “Performance-oriented model learning for data-driven MPC design”. *IEEE Control Systems Letters*, **3**(3), pp. 577–582.
 - [21] Soroufifar, F., Makrygiorgos, G., Mesbah, A., and Paulson, J. A., 2021. “A data-driven automatic tuning method for MPC under uncertainty using constrained Bayesian optimization”. *IFAC-PapersOnLine*, **54**(3), pp. 243–250.
 - [22] Lu, Q., González, L. D., Kumar, R., and Zavala, V. M., 2021. “Bayesian optimization with reference models: A case study in MPC for HVAC central plants”. *Computers & Chemical Engineering*, **154**, p. 107491.
 - [23] Paulson, J. A., Makrygiorgos, G., and Mesbah, A., 2022. “Adversarially robust Bayesian optimization for efficient auto-tuning of generic control structures under uncertainty”. *AIChE Journal*, **68**(6), p. e17591.
 - [24] Krishnamoorthy, D., and Doyle, F. J., 2022. “Safe Bayesian optimization using interior-point method - applied to personalized insulin dose guidance”. *IEEE Control Systems Letters*, **6**, pp. 2834–2839.
 - [25] Berkenkamp, F., Krause, A., and Schoellig, A. P., 2021. “Bayesian optimization with safety constraints: safe and automatic parameter tuning in robotics”. *Machine Learning*, pp. 1–35.
 - [26] Paulson, J. A., Soroufifar, F., Laughman, C. R., and Chakrabarty, A., 2023 (accepted). “LSR-BO: Local search region constrained Bayesian optimization for performance optimization of vapor compression systems”. In *Proceedings of American Control Conference*, IEEE.
 - [27] Williams, C. K., and Rasmussen, C. E., 2006. *Gaussian Processes For Machine Learning*, Vol. 2. MIT press Cambridge, MA.
 - [28] Jones, D. R., Schonlau, M., and Welch, W. J., 1998. “Efficient global optimization of expensive black-box functions”. *Journal of Global optimization*, **13**(4), pp. 455–492.
 - [29] Gardner, J., Kusner, M., et al., 2014. “Bayesian optimization with inequality constraints”. In *Proc. of the 31st International Conference on Machine Learning (ICML)*, Vol. 32 of *Proceedings of Machine Learning Research*, PMLR, pp. 937–945.
 - [30] Nocedal, J., and Wright, S. J., 2006. *Numerical Optimization*, 2e ed. Springer, New York, NY, USA.
 - [31] Rasmussen, C. E., and Williams, C. K. I., 2006. *Gaussian processes for machine learning*. MIT Press, Cambridge, MA.
 - [32] Gelbart, M. A., 2015. “Constrained Bayesian Optimization and Applications”. Ph.D. dissertation, Harvard University.
 - [33] Letham, B., Karrer, B., Ottoni, G., and Bakshy, E., 2019. “Constrained Bayesian optimization with noisy experiments”.
 - [34] Chowdhury, S. R., and Gopalan, A., 2017. “On kernelized multi-armed bandits”. In *International Conference on Machine Learning*, PMLR, pp. 844–853.
 - [35] Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. W., 2012. “Information-theoretic regret bounds for Gaussian process optimization in the bandit setting”. *IEEE Transactions on Information Theory*, **58**(5), pp. 3250–3265.
 - [36] Berkenkamp, F., Schoellig, A. P., and Krause, A., 2019. “No-regret Bayesian optimization with unknown hyperparameters”. *Journal of Machine Learning Research*, **20**, pp. 1–24.
 - [37] Vazquez, E., and Bect, J., 2010. “Convergence properties of the expected improvement algorithm with fixed mean and covariance functions”. *Journal of Statistical Planning and inference*, **140**(11), pp. 3088–3095.
 - [38] Bull, A. D., 2011. “Convergence rates of efficient global optimization algorithms”. *Journal of Machine Learning Research*, **12**(10).
 - [39] Lu, C., and Paulson, J. A., 2022. “No-regret bayesian optimization with unknown equality and inequality constraints using exact penalty functions”. *IFAC-PapersOnLine*, **55**(7), pp. 895–902.
 - [40] Xu, W., Jiang, Y., and Jones, C. N., 2022. “Constrained efficient global optimization of expensive black-box functions”. *arXiv preprint arXiv:2211.00162*.
 - [41] Balandat, M., Karrer, B., Jiang, D., Daulton, S., Letham, B., Wilson, A. G., and Bakshy, E., 2020. “BoTorch: a framework for efficient Monte-Carlo Bayesian optimization”. *Advances in neural information processing systems*, **33**, pp. 21524–21538.
 - [42] Liu, D. C., and Nocedal, J., 1989. “On the limited memory BFGS method for large scale optimization”. *Mathematical Programming*, **45**(1), pp. 503–528.
 - [43] Paulson, J. A., Soroufifar, F., and Chakrabarty, A., 2022.

- “Efficient multi-step lookahead bayesian optimization with local search constraints”. In Proceedings of the Conference on Decision and Control, IEEE, pp. 123–129.
- [44] Eriksson, D., et al., 2019. “Scalable global optimization via local Bayesian optimization”. In Advances in Neural Information Processing Systems, pp. 5496–5507.
- [45] Modelica Association, 2017. “Modelica specification, Version 3.4”.
- [46] Dassault Systemes, 2019. “Dymola 2020”.
- [47] Modelica Association, 2019. “Functional Mockup Interface for Model Exchange and Co-Simulation, Version 2.0.1”.