



Enhanced Bayesian Optimization via Preferential Modeling of Abstract Properties

A. V. Arun Kumar^(✉), Alistair Shilton, Sunil Gupta, Santu Rana,
Stewart Greenhill, and Svetha Venkatesh

Applied Artificial Intelligence Institute (A²I²), Deakin University, Geelong, Australia
a.anjanapuravenkatesh@deakin.edu.au

Abstract. Experimental (design) optimization is a key driver in designing and discovering new products and processes. Bayesian Optimization (BO) is an effective tool for optimizing expensive and black-box experimental design processes. While Bayesian optimization is a principled data-driven approach to experimental optimization, it learns everything from scratch and could greatly benefit from the expertise of its human (domain) experts who often reason about systems at different abstraction levels using physical properties that are not necessarily directly measured (or measurable). In this paper, we propose a human-AI collaborative Bayesian framework to incorporate expert preferences about unmeasured abstract properties into the surrogate modeling to further boost the performance of BO. We provide an efficient strategy that can also handle any incorrect/misleading expert bias in preferential judgments. We discuss the convergence behavior of our proposed framework. Our experimental results involving synthetic functions and real-world datasets show the superiority of our method against the baselines.

Keywords: Machine Learning · Bayesian Optimization · Gaussian Process · Expert Feedback · Preferential Modeling

1 Introduction

Experimental design is the workhorse of scientific design and discovery. Bayesian Optimization (BO) has emerged as a powerful methodology for experimental design tasks [1, 2] due to its sample-efficiency in optimizing expensive black-box functions. In its basic form, BO starts with a set of randomly initialized designs and then sequentially suggests the next design until the target objective is reached or the optimization budget is depleted. Theoretical analyses [3, 4] of BO methods have provided mathematical guarantees of sample efficiency in the form of sub-linear regret bounds. While BO is an efficient optimization method, it only uses data gathered during the design optimization process. However, in real world

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-70365-2_14.

experimental design tasks, we also have access to human experts [5] who have enormous knowledge about the underlying physical phenomena. Incorporating such valuable knowledge can greatly accelerate the sample-efficiency of BO.

Previous efforts in BO literature have incorporated expert knowledge on the shape of functions [6], form of trends [7], priors over optima [8] and model selection [9], which require experts to provide very detailed knowledge about the black-box function. However, most experts understand the process in an approximate or qualitative way, and usually reason in terms of the intermediate abstract properties - the expert will compare designs, and reason as to why one design is better than another using high level abstractions. For instance, consider the design of a spacecraft shield (Whipple shield) consisting of 2 plates separated by a gap to safeguard the spacecraft against micro-meteoroid and orbital debris particle impacts. The design efficacy is measured by observing the penetration caused by hyper-velocity debris. An expert would reason why one design is better than another and accordingly come up with a new design to try out. As part of their domain knowledge, human experts often expect the first plate to shatter the space debris while the second to absorb the fragments effect. Based on these abstract intuitions, the expert will compare a pair of designs by examining the shield penetration images and ask: Does the first plate shatter better (*Shattering*)? Does the second plate absorb the fragments better (*Absorption*)? The use of such abstractions allows experts to predict the overall design objective thus resulting in an efficient experimental design process. It is important to note that measuring such abstractions is not usually feasible and only expert’s qualitative inputs are available. *Incorporating such abstract properties in BO for the acceleration of experimental design process is not well explored.*

In this paper, we propose a novel human-AI collaborative approach - **Bayesian Optimization with Abstract Properties (BOAP)** - to accelerate BO by capturing expert inputs about the abstract, unmeasurable properties of the designs. Since expert inputs are usually qualitative [10] and often available in the form of design preferences based on abstract properties, we model each abstract property via a latent function using the qualitative pairwise rankings. We note that eliciting such pairwise preferences about designs does not add significant cognitive overhead for the expert, in contrast to asking for explicit knowledge about properties. We fit a separate rank Gaussian process [11] to model each property. Our framework allows enormous flexibility for expert collaborations as it does not need the exact value of an abstract property, just its ranking. A schematic of our proposed BOAP framework is shown in Fig. 1.

Although we anticipate that experts will provide accurate preferences on abstract properties, the expert preferential knowledge can sometimes be misleading. Therefore to avoid such undesired bias, we use two models for the black-box function. The first model uses a “main” Gaussian Process (GP) to model the black-box function in an augmented input space where the design variables are augmented with the estimated abstract properties modeled via their respective rank GPs. The second model uses another “main” GP to model the black-box function using the original design space *without* any expert inputs. At each iteration, we use predictive likelihood-based model selection to choose the “best” model that has higher probability of finding the optima.

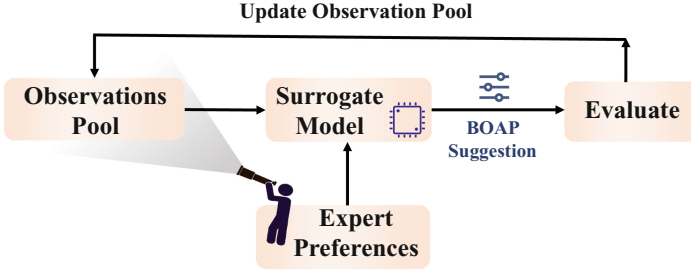


Fig. 1. A schematic representation of our proposed framework Bayesian Optimization with Abstract Properties (BOAP).

Our contributions are: **(i)** we propose a novel human-AI collaborative BO algorithm (BOAP) for incorporating the expert pairwise preferences on abstract properties via rank GPs (Sect. 3), **(ii)** we provide a brief discussion on the convergence behavior of our proposed BOAP method (Sect. 4), **(iii)** we provide empirical results on both synthetic optimization problems and real-world design optimization problems to prove the usefulness of BOAP framework (Sect. 5).

2 Background

Notations

We use lower case bold fonts \mathbf{v} for vectors and v_i for each element in \mathbf{v} . \mathbf{v}^\top is the transpose. We use upper case bold fonts (and bold greek symbols) \mathbf{M} for matrices and M_{ij} for each element in \mathbf{M} . $\text{abs}(\cdot)$ is the absolute value. $|\cdot|$ is the determinant. $\mathbb{N}_n = \{1, 2, \dots, n\}$. \mathbb{R} for Reals. \mathcal{X} is a index set and $\mathbf{x} \in \mathcal{X}$.

2.1 Bayesian Optimization

Bayesian Optimization (BO) [12, 13] provides an elegant framework for finding the global optima of an expensive black-box function $f(\mathbf{x})$, given as $\mathbf{x}^* \in \text{argmax}_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x})$, where \mathcal{X} is a compact search space. BO is comprised of two main components: (i) a surrogate model (usually a Gaussian Process [11]) of the unknown objective function $f(\mathbf{x})$, and (ii) an Acquisition Function $u(\mathbf{x})$ [14] to guide the search for optima.

Gaussian Process. A Gaussian Process (GP) [11] is a flexible, non-parametric distribution over functions. It is a preferred surrogate model because of its simplicity and tractability, in contrast to other surrogate models such as Student-t process [15] and Wiener process [16]. A GP is defined by a prior mean function $\mu(\mathbf{x})$ and a kernel $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$. The function $f(\mathbf{x})$ is modeled using a GP as $f(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}'))$. If $\mathcal{D}_{1:t} = \{\mathbf{x}_{1:t}, \mathbf{y}_{1:t}\}$ denotes a set of observations, where $y = f(\mathbf{x}) + \eta$ is the observation corrupted with noise

$\eta \in \mathcal{N}(0, \sigma_\eta^2)$ then, according to the properties of GP, the observed samples $\mathcal{D}_{1:t}$ and a new observation $(\mathbf{x}_*, f(\mathbf{x}_*))$ are jointly Gaussian. Thus, the posterior distribution $f(\mathbf{x}_*)$ is $\mathcal{N}(\mu(\mathbf{x}_*), \sigma^2(\mathbf{x}_*))$, where $\mu(\mathbf{x}_*) = \mathbf{k}^\top [\mathbf{K} + \sigma_\eta^2 \mathbf{I}]^{-1} \mathbf{y}_{1:t}$, $\sigma^2(\mathbf{x}_*) = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}^\top [\mathbf{K} + \sigma_\eta^2 \mathbf{I}]^{-1} \mathbf{k}$, $\mathbf{k} = [k(\mathbf{x}_*, \mathbf{x}_1) \cdots k(\mathbf{x}_*, \mathbf{x}_t)]^\top$, and $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)]_{i,j \in \mathbb{N}_t}$.

Acquisition Functions. The acquisition function selects the next point for evaluation by balancing the exploitation vs exploration (i.e. searching in high value regions vs highly uncertain regions). Some popular acquisition functions include Expected Improvement (EI) [17], GP-UCB [3] and Thompson Sampling (TS) [18]. A standard BO algorithm is provided in Sect. 8 of the supplementary material¹.

2.2 Rank GP Distributions

[19] demonstrated that humans are better at providing qualitative comparisons than absolute magnitudes. Thus modeling latent human preferences is crucial when optimization objectives are in domains such as A/B testing of web designing [20], recommender systems [21], players skill rating [22] and many more. [23] proposed a non-parametric Bayesian algorithm for learning instance or label preferences. We now discuss modeling pairwise preference relations using rank GPs.

Consider a set of n distinct training instances denoted by $X = \{\mathbf{x}_i \forall i \in \mathbb{N}_n\}$ based on which pairwise preference relations are observed. Let $P = \{(\mathbf{x} \succ \mathbf{x}') \mid \mathbf{x}, \mathbf{x}' \in X\}$ be a set of pairwise preference relations, where the notation $\mathbf{x} \succ \mathbf{x}'$ expresses the preference of instance \mathbf{x} over \mathbf{x}' . For example, the pair $\{\mathbf{x}, \mathbf{x}'\}$ can be two different spacecraft shield designs and $\mathbf{x} \succ \mathbf{x}'$ implies that spacecraft design \mathbf{x} is preferred over \mathbf{x}' . [23] assume that each training instance is associated with an unobservable latent function value $\{\bar{f}(\mathbf{x})\}$ measured from an underlying hidden preference function $\bar{f} : \mathbb{R}^d \rightarrow \mathbb{R}$, where $\mathbf{x} \succ \mathbf{x}'$, implies $\bar{f}(\mathbf{x}) > \bar{f}(\mathbf{x}')$. Employing an appropriate GP prior and likelihood, user preference can be modeled via rank Gaussian process distributions.

Preference learning has been used in BO literature [24, 25]. [24] proposed Preferential BO (PBO) to model the unobserved objective function using a binary design preferential feedback. [26] modified PBO to compute posteriors via skew GPs. [27] proposed a preference learning based BO to model preferences in a multi-objective setup using multi-output GPs. [28] proposed a preference learning with Siamese Networks to capture preferences in a Multi-task learning setup. All these works incorporate preferences about an unobserved objective function. However, in this paper, we use preference learning to model expert preferences about the intermediate abstract (auxiliary) properties. Our latent model learned using such preferential data is then used as an input to model the main objective function.

¹ The supplementary material of BOAP is accessible online at the following link: https://doi.org/10.1007/978-3-031-70365-2_14.

3 Framework

This paper addresses the global optimization of an expensive, black-box function f , *i.e.*, we aim to find the global optima (\mathbf{x}^*) of the unknown objective function f represented as:

$$\mathbf{x}^* \in \operatorname{argmax}_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) \quad (1)$$

where $f : \mathcal{X} \rightarrow \mathbb{R}$ is a noisy and expensive objective function. For example, f could be a metric signifying the strength of the spacecraft shield. The motivation of this research work is to model f by capturing the cognitive knowledge of experts in making preferential decisions based on the inherent non-measurable abstract properties of the possible designs. The objective here is same as that of standard BO *i.e.*, to find the optimal design (\mathbf{x}^*) that maximizes the unknown function f , but in the light of expert preferential knowledge on abstract properties. The central idea is to use preferential feedback to model and utilize the underlying higher-order properties that underpin preferential decisions about designs. We propose **Bayesian Optimization with Abstract Properties (BOAP)** for the optimization of f in the light of expert preferential inputs. First, we discuss expert knowledge about abstract properties. Next, we discuss GP modeling of f with preferential inputs, followed by a model-selection step that is capable of overcoming a futile expert bias in preferential knowledge. A complete algorithm for BOAP is presented in Algorithm 1 at the end of this section.

3.1 Expert Preferential Inputs on Abstract Properties

In numerous scenarios, domain experts reason the output of a system in terms of higher-order properties $\omega_1(\mathbf{x}), \omega_2(\mathbf{x}), \dots$ of a design $\mathbf{x} \in \mathcal{X}$. However, these abstract properties are rarely measured, only being accessible via expert preferential inputs. For instance, a material scientist designing spacecraft shield can easily provide her pairwise preferences on the properties such as shattering, shock absorption, *i.e.*, “*this design absorbs shock better than that design*”, in contrast to specifying the exact measurements of shock absorption. These properties can be simple physical properties or abstract combinations of multiple physical properties which an expert uses to reason about the output of a system. We propose to incorporate such qualitative properties accessible to the expert in the surrogate modeling of the given objective function to further accelerate the sample-efficiency of BO.

Let $\omega_{1:m}(\mathbf{x})$ be a set of m abstract properties derived from the design $\mathbf{x} \in \mathcal{X}$. For property ω_i , design \mathbf{x} is preferred over design \mathbf{x}' if $\omega_i(\mathbf{x}) > \omega_i(\mathbf{x}')$. We denote the set of preferences provided on ω_i as $P^{\omega_i} = \{(\mathbf{x} \succ \mathbf{x}') \text{ if } \omega_i(\mathbf{x}) > \omega_i(\mathbf{x}') \mid \mathbf{x} \in \mathcal{X}\}$.

Rank GPs for Abstract Properties. We capture the aforementioned expert preferential data for each of the abstract properties $\omega_{1:m}$ individually using m separate rank Gaussian process distributions [23]. In conventional GPs the observation model consists of a map of input-output pairs. In contrast, the observation

model of a rank (preferential) Gaussian process (\mathcal{GP}) consists of a set of instances and a set of pairwise preferences between those instances. The central idea here is to capture the ordering over a set of n instances $X = \{\mathbf{x}_i | \forall i \in \mathbb{N}_n\}$ by learning latent preference functions $\{\omega_i | \forall i \in \mathbb{N}_m\}$. We denote such a rank GP modeling abstract property ω_i by the notation \mathcal{GP}_{ω_i} .

Let $\mathcal{X} \in \mathbb{R}^d$ be a d -dimensional compact search space and $X = \{\mathbf{x}_i | \forall i \in \mathbb{N}_n\}$ be a set of n training instances. Let $\omega = \{\omega(\mathbf{x})\}$ be the unobservable latent preference function values associated with each of the instances $\mathbf{x} \in X$. Let P be the set of p pairwise preferences between instances in X , defined as:

$$P = \{(\mathbf{x} \succ \mathbf{x}')_j \text{ if } \omega(\mathbf{x}) > \omega(\mathbf{x}') \mid \mathbf{x} \in X, \forall j \in \mathbb{N}_p\}$$

where ω is the latent preference function. The observation model for the rank GP distribution \mathcal{GP}_ω modeling the latent preference function ω is given as:

$$\bar{\mathcal{D}} = \{\mathbf{x}_{1:n}, P = \{(\mathbf{x} \succ \mathbf{x}')_j \forall \mathbf{x}, \mathbf{x}' \in X, j \in \mathbb{N}_p\}\}$$

We follow the probabilistic kernel approach for preference learning [23] to formulate the likelihood function and Bayesian probabilities. Imposing non-parametric GP priors on the latent function values ω , we arrive at the prior probability of ω given by:

$$\mathcal{P}(\omega) = (2\pi)^{-\frac{n}{2}} |\mathbf{K}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2} \omega^\top \mathbf{K}^{-1} \omega\right) \quad (2)$$

With suitable noise assumptions $\mathcal{N}(0, \tilde{\sigma}_\eta^2)$ on inputs and the preference relations $(\mathbf{x}, \mathbf{x}')_{1:p}$ in P , the Gaussian likelihood function based on [29] is:

$$\mathcal{P}((\mathbf{x} \succ \mathbf{x}')_i | \omega(\mathbf{x}), \omega(\mathbf{x}')) = \Phi(z_i(\mathbf{x}, \mathbf{x}')) \quad (3)$$

where Φ is the c.d.f of standard normal distribution and $z(\mathbf{x}, \mathbf{x}') = \frac{\omega(\mathbf{x}) - \omega(\mathbf{x}')}{\sqrt{2\tilde{\sigma}_\eta^2}}$. Based on Bayes theorem, the posterior distribution of the latent function given the data is given by:

$$\mathcal{P}(\omega | \bar{\mathcal{D}}) = \frac{\mathcal{P}(\omega)}{\mathcal{P}(\bar{\mathcal{D}})} \mathcal{P}(\bar{\mathcal{D}} | \omega)$$

where $\mathcal{P}(\omega)$ is the prior distribution (Eq. (2)), $\mathcal{P}(\bar{\mathcal{D}}) = \int \mathcal{P}(\bar{\mathcal{D}} | \omega) \mathcal{P}(\omega) d\omega$ is the evidence of model parameters including kernel hyperparameters, and $\mathcal{P}(\bar{\mathcal{D}} | \omega)$ is the probability of observing the pairwise preferences given the latent function values ω , which can be computed as a product of the likelihood (Eq. (3)) *i.e.*, $\mathcal{P}(\bar{\mathcal{D}} | \omega) = \prod_p \mathcal{P}((\mathbf{x} \succ \mathbf{x}')_p | \omega(\mathbf{x}), \omega(\mathbf{x}'))$. We find the posterior distribution using Laplace approximation and the Maximum A Posteriori estimate (MAP) ω_{MAP} as the mode of posterior distribution. We can find the MAP using Newton-Raphson descent given by:

$$\omega^{\text{new}} = \omega^{\text{old}} - \mathbf{H}^{-1} \mathbf{g}|_{\omega=\omega^{\text{old}}} \quad (4)$$

where the Hessian $\mathbf{H} = [\mathbf{K} + \tilde{\sigma}_\eta^2 \mathbf{I}]^{-1} + \mathbf{C}$, and the gradient $\mathbf{g} = \nabla_\omega \log \mathcal{P}(\omega | \bar{\mathcal{D}}) = -[\mathbf{K} + \tilde{\sigma}_\eta^2 \mathbf{I}]^{-1} \omega + \mathbf{b}$, given $b_j = \frac{\partial}{\partial \omega(\mathbf{x}_j)} \sum_p \ln \Phi(z_p)$ and $C_{ij} = \frac{-\partial^2}{\partial \omega(\mathbf{x}_i) \partial \omega(\mathbf{x}_j)} \sum_p \ln \Phi(z_p)$.

Hyperparameter Optimization. Kernel hyperparameters (θ) are crucial to optimize the generalization performance of the GP. We perform the model selection for our rank-GPs by maximizing the corresponding log-likelihood in the light of latent values $\boldsymbol{\omega}_{\text{MAP}}$. In contrast to the evidence maximization mentioned in [23] *i.e.*, $\theta_{\omega}^* = \operatorname{argmax}_{\theta_{\omega}} \mathcal{P}(\mathcal{D}|\theta_{\omega})$, we find the optimal kernel hyperparameters by maximizing the log-likelihood ($\bar{\mathcal{L}}$) of rank GPs *i.e.*, $\theta_{\omega}^* = \operatorname{argmax}_{\theta_{\omega}} \bar{\mathcal{L}}$. The closed-form of log-likelihood of the rank GP is given as:

$$\bar{\mathcal{L}} = -\frac{1}{2} \boldsymbol{\omega}_{\text{MAP}}^{\top} [\mathbf{K} + \tilde{\sigma}_{\eta}^2 \mathbf{I}]^{-1} \boldsymbol{\omega}_{\text{MAP}} - \frac{1}{2} \log |\mathbf{K} + \tilde{\sigma}_{\eta}^2 \mathbf{I}| - \frac{n}{2} \log(2\pi) \quad (5)$$

3.2 Augmented GP with Abstract Property Preferences

To account for property preferences in modeling f , we augment the input \mathbf{x} of a conventional GP modeling f with the mean predictions obtained from m rank GPs ($\mathcal{GP}_{\omega_{1:m}}$) as auxiliary inputs capturing the property preferences $\omega_{1:m}$, in other words, instead of modeling GP directly on \mathbf{x} we model on $\tilde{\mathbf{x}} = [\mathbf{x}, \mu_{\omega_1}(\mathbf{x}), \dots, \mu_{\omega_m}(\mathbf{x})]$, where μ_{ω_i} is the predictive mean computed using:

$$\mu_{\omega_i}(\mathbf{x}) = \mathbf{k}^{\top} [\mathbf{K} + \sigma_{\eta}^2 \mathbf{I}]^{-1} \boldsymbol{\omega}_{\text{MAP}}$$

where $\mathbf{k} = [k(\mathbf{x}, \mathbf{x}_1), \dots, k(\mathbf{x}, \mathbf{x}_n)]^{\top}$, $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)]_{i,j \in \mathbb{N}_n}$ and $\mathbf{x}_i \in X$. To handle different scaling levels in rank GPs, we normalize its output in the interval $[0, 1]$, such that $\mu_{\omega_i}(\mathbf{x}) \in [0, 1]$.

Although we model $\tilde{\mathbf{x}}$ using mean predictions $\mu_{\omega_i}(\mathbf{x})$, the uncertainty estimates were not (directly) considered in the modeling. The GP predictive variance tends to be high outside of the neighborhood of observations, indicating the uncertainty in our beliefs on the model. Therefore, a data point with high predictive variance $(\sigma_{\omega_i}(\mathbf{x}))^2$ in rank GP indicates the model uncertainty. We incorporate this uncertainty in our main GP modeling $\tilde{\mathbf{x}}$ such that the effects of predicted abstract properties $\mu_{\omega_i}(\mathbf{x})$ are appropriately reduced when the model is uncertain *i.e.* when $(\sigma_{\omega_i}(\mathbf{x}))^2$ is high.

To achieve this, we formulate the feature-wise lengthscales as a function of predictive uncertainty of the augmented dimensions to control their *importance* in the overall GP. Note that augmented features can be detrimental when the model is uncertain. To address this potential problem, we use a spatially varying kernel [6] that treats the lengthscale as a function of the input, rather than a constant. A positive definite kernel with spatially varying lengthscale is given as:

$$k(\mathbf{x}, \mathbf{x}') = \prod_{i=1}^d \sqrt{\frac{2l(x_i)l(x'_i)}{l^2(x_i) + l^2(x'_i)}} \exp\left(-\sum_{i=1}^d \frac{(x_i - x'_i)^2}{l^2(x_i) + l^2(x'_i)}\right) \quad (6)$$

where $l(\cdot)$ is the lengthscale function and $\mathbf{x} \in \mathbb{R}^d$. In our proposed framework, we model $\tilde{\mathbf{x}} \in \mathbb{R}^{d+m}$ and use lengthscale as a function $l(\cdot)$ only for the newly augmented (m) dimensions and retain the lengthscales of the original (d) dimensions to standard constant values *i.e.* $l(x_i) = l_i \forall i \in \mathbb{N}_d$. Therefore the overall kernel

hyperparameter set is given as $\theta = [l_1, \dots, l_d, l_{\omega_1}(\mathbf{x}), \dots, l_{\omega_m}(\mathbf{x})]$. As we need lengthscale function to reflect the model uncertainty, we set $l_{\omega_i}(\mathbf{x}) = \alpha_i \tilde{\sigma}_{\omega_i}(\mathbf{x})$, where $\tilde{\sigma}_{\omega_i}(\mathbf{x})$ is the normalized standard deviation of the rank GP predicted for the abstract property ω_i and α_i is a scale parameter that is tuned using the standard GP log-marginal likelihood in conjunction with other kernel parameters. The aforementioned lengthscales ensure that the data points $\tilde{\mathbf{x}}$ with high model uncertainty have higher lengthscale on the augmented dimensions and thus are treated as less important.

The objective function is modeled on the concatenated inputs $\tilde{\mathbf{x}} \in \mathbb{R}^{d+m}$ using the spatially varying kernel (Eq. (6)) $k(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}) \forall \tilde{\mathbf{x}} \in \mathbb{R}^{d+m}$ and we denote this function with augmented inputs $\tilde{\mathbf{x}}$ as human-inspired objective function $h(\tilde{\mathbf{x}})$. The GP (\mathcal{GP}_h) constructed in the light of expert preferential data is then used in BO to find the global optima of $h(\tilde{\mathbf{x}})$, given as:

$$\mathbf{x}^* \in \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmax}} h(\tilde{\mathbf{x}}) \quad (7)$$

The observation model is $\mathcal{D} = \{(\mathbf{x}, y = h(\tilde{\mathbf{x}}) \approx f(\mathbf{x}))\}$ *i.e.* the human-inspired objective function $h(\tilde{\mathbf{x}})$ is a simplified $f(\mathbf{x})$ with auxiliary features in the input, thus we observe the $h(\tilde{\mathbf{x}})$ via $f(\mathbf{x})$. The kernel hyperparameters associated with \mathcal{GP}_h are denoted as θ_h given as $\theta_h = \{l_{1:d}, \alpha_{1:m}\}$.

3.3 Overcoming Inaccurate Expert Inputs

Up to this point we have assumed that expert input is accurate and thus likely to accelerate BO. However, in some cases this feedback may be inaccurate, and potentially slowing optimization. To overcome such bias and encourage exploration we maintain 2 models, one of which is augmented by expert abstract properties (we refer to this as Human Arm-h) and an un-augmented model (we refer to this as Control Arm-f), and use predictive likelihood to select the arm at each iteration.

The control arm models f directly by observing the function values at suggested candidate points. Here, we fit a standard GP (\mathcal{GP}_f) based on the data collected *i.e.*, $\mathcal{D} = \{(\mathbf{x}, y = f(\mathbf{x}) + \eta)\}$ where $\eta \sim \mathcal{N}(0, \sigma_\eta^2)$ is the Gaussian noise. The GP distribution (\mathcal{GP}_f) with hyperparameters $\theta_f = \{l_{1:d}\}$ may be used to optimize f using a BO algorithm.

At each iteration t , we compare the predictive likelihoods (\mathcal{L}_t) of both the human augmented arm (Arm-h) and the control arm (Arm-f) to select the arm to pull for suggesting the next promising candidate for the function evaluation. Then, we use Thompson Sampling (TS) strategy [18] to draw a sample S_t from the GP distribution of the arm pulled and find its corresponding maxima given as:

$$\mathbf{x}_t^h = \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmax}} (S^h(\tilde{\mathbf{x}})); \quad \mathbf{x}_t^f = \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmax}} (S^f(\mathbf{x})) \quad (8)$$

The arm with maximum predictive likelihood is chosen at each iteration and we observe f at the suggested location *i.e.*, $(\mathbf{x}_t^h, f(\mathbf{x}_t^h))$ or $(\mathbf{x}_t^f, f(\mathbf{x}_t^f))$. Then rank GPs are updated to capture the preferences with respect to the new suggestion

\mathbf{x}_t^h or \mathbf{x}_t^f . This process continues until the evaluation budget T is exhausted. A complete flowchart of our framework is shown in Fig. 2. Additional details of BOAP framework are provided in the supplementary material (Sect. 9).

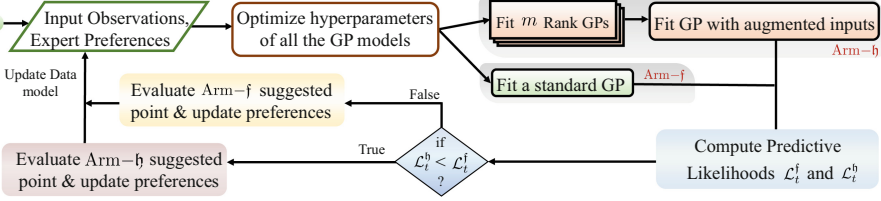


Fig. 2. A complete process flowchart of our proposed BOAP framework.

Algorithm 1. BO with Preferences on Abstract Properties (BOAP)

Input: Sampling Budget T , Initial Samples: $\mathcal{D}_{1:t'} = \{\mathbf{x}_{1:t'}, \mathbf{y}_{1:t'}\}$, Expert Preferences: $P^{\omega_i} = \{(\mathbf{x} \succ \mathbf{x}')_{1:p} \mid \forall i \in \mathbb{N}_m\}$

1. **for** $t = t' + 1, \dots, T$ iterations **do**
 2. optimize hyperparameters $\Theta_t^* = \{\theta_{\omega_{1:m}}^*, \theta_h^*, \theta_f^*\}$ and update $\mathcal{GP}_{\omega_{1:m}}, \mathcal{GP}_h, \mathcal{GP}_f$
 3. compute predictive likelihoods \mathcal{L}_t^h and \mathcal{L}_t^f for Arm-h and Arm-f
 4. **if** $\mathcal{L}_t^h > \mathcal{L}_t^f$, **then**
 5. draw a random sample S_t^h from Arm-h using Thompson Sampling
 6. maximize S_t^h to find $\mathbf{x}_t^h = \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmax}} (S_t^h(\tilde{\mathbf{x}}))$
 7. $\mathbf{x}_t = \mathbf{x}_t^h$
 8. **else,**
 9. draw a random sample S_t^f from Arm-f using Thompson Sampling
 10. maximize S_t^f to find $\mathbf{x}_t^f = \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmax}} (S_t^f(\mathbf{x}))$
 11. $\mathbf{x}_t = \mathbf{x}_t^f$
 12. evaluate f at \mathbf{x}_t to obtain $y_t = f(\mathbf{x}_t) + \eta_t$
 13. augment data $\mathcal{D} = \mathcal{D} \cup (\mathbf{x}_t, y_t)$ and update expert preferences $P^{\omega_{1:m}}$ w.r.t \mathbf{x}_t
 14. $\mathbf{x}^* = \underset{(\mathbf{x}, y) \in \mathcal{D}}{\operatorname{argmax}} y$
 15. **end for**
 16. return \mathbf{x}^*
-

4 Convergence Remarks

In this section we discuss the convergence of our BOAP algorithm in terms of regret bounds. As we are dealing with human expert feedback in our algorithm, it is difficult to make absolute statements as we are reliant on the accuracy of the feedback given and the knowledge of the expert involved, which may be

limited if the objective must explore less thoroughly understood areas of the search space (so the expert learns alongside the GP model). Nevertheless, with minimal assumptions we may draw some conclusions that help us to better understand the impact of expert feedback, which is important not only to better understand the potential of BOAP to accelerate convergence but also to give insight into possible future directions.

BOAP may be understood as kernel learning in practice - the core distinction between the human and control arms is that the human arm features an evolving kernel (6). Assuming for simplicity that the human arm comes to dominate over time (as measured by likelihood) then the influence of the kernel (6) on the convergence of the BO algorithm is measured through the maximum information gain $\gamma_T(d)$, where d is the input dimension. For Thompson sampling type algorithm the cumulative regret $R_T = \sum_t f(\mathbf{x}^*) - f(\mathbf{x}_t)$ is typically [4, 30] bounded as $R_T = \mathcal{O}(\sqrt{T\gamma_T(d)})$ (up to log factors), where the maximum information gain $\gamma_T(d)$ is governed by the kernel K through the eigenvalues $\lambda_1 \geq \lambda_2 \geq \dots$ of the covariance matrix \mathbf{K}_T evaluated on the observations $\{(\mathbf{x}_t, y_t) : t \leq T\}$ [3]:

$$\gamma_T(d) \leq \frac{\frac{1}{2}}{1 - \frac{1}{e}} \max_{(m_t: \sum_t m_t = T)} \sum_{t=1}^{|\mathcal{D}|} \log(1 + \sigma^{-2} m_t \lambda_t) \quad (9)$$

Moreover in [3] it is shown that the asymptotic behavior of $\gamma_T(d)$ is controlled by the dimension d of the input. Our key insight for the kernel (6) is that we can drop features with lengthscales over a threshold without overly perturbing the kernel, effectively replacing d with the number of features (d_{eff}) having lengthscales below the threshold, bounding the resulting error so introduced.

Assuming that **(a)** expert observations obey a simple convergence assumption $\max_{\mathbf{x} \in \mathcal{X}, t} K_{\omega_i}(\mathbf{x}, \mathbf{x}_t) = \mathcal{O}(g(T))$, where $g(T) \rightarrow 0$ as $T \rightarrow \infty$, and **(b)** as $T \rightarrow \infty$, only $d_{\text{eff}} < d$ of the lengthscales (augmenting or otherwise) satisfy $l_d, l_{\omega_i} < l_{\text{max}}$, then, for the kernel (6), $\gamma_T(d) \leq \check{\gamma}_T(d_{\text{eff}}) + \mathcal{O}(\frac{d_{\text{eff}}}{l_{\text{max}}^2}) + \mathcal{O}(g(T))$. In this expression $\check{\gamma}_T(d_{\text{eff}})$ is the maximum information gain for a d_{eff} dimensional SE kernel, *i.e.* [3] $\check{\gamma}_T(d_{\text{eff}}) = \mathcal{O}((\log T)^{d_{\text{eff}}+1})$. Thus we would expect cumulative regret to satisfy:

$$R_T = \mathcal{O}\left(\sqrt{T\left((\log T)^{d_{\text{eff}}+1} + \frac{d_{\text{eff}}}{l_{\text{max}}^2}\right)}\right) \quad (10)$$

That is, the regret bound for BOAP, assuming the human arm dominates as $T \rightarrow \infty$, is the the regret bound for BO with effective dimension d_{eff} plus a term that scales as the ratio of d_{eff} and the cut-off lengthscale l_{max}^2 . The more effectively the augmenting features are able to summarize the data in a useful way that renders other features superfluous (*i.e.* minimizes d_{eff}), the tighter the regret bound becomes. A detailed discussion on the maximum information gain and the regret bounds is provided in the supplementary material (Sect. 10)

5 Experiments

We evaluate the performance of BOAP method using synthetic benchmark function optimization problems and real-world optimization problems arising

in advanced battery manufacturing processes. We have considered the following experimental settings for BOAP. We use the popular Automatic Relevance Determination (ARD) kernel [31] for the construction of both the rank GPs and the conventional (un-augmented) GPs. For rank GPs, we tune ARD kernel hyperparameters $\theta_d = \{l_d\}$ using max-likelihood estimation (Eq. (5)). For the augmented GP modeling $\tilde{\mathbf{x}}$, we use a spatially varying kernel with a parametric lengthscale function (See discussion in Sect. 3.2). As we normalize the bounds, we tune l_d (the lengthscale for the *un*-augmented features) in the interval $[0.1, 1]$ and the scale parameter α (for the auxiliary features) in the interval $(0, 2]$. Further, we set signal variance $\sigma_f^2 = 1$ as we standardize the outputs.

We compare the performance of BOAP algorithm with the following state-of-the-art baselines. (i) **BO-TS**: a standard Bayesian Optimization (BO) with Thompson Sampling (TS) strategy, (ii) **BO-EI**: BO with Expected Improvement (EI) acquisition function, and (iii) BOAP - Only Augmentation (**BOAP-OA**): Here we run our algorithm without the 2-arm scheme and we only use augmented input for GP modeling. This method shows the effectiveness of expert’s inputs. We evaluate the performance of our method against the baselines by plotting the simple regret (\mathcal{R}_t) given by: $\mathcal{R}_t = f(\mathbf{x}^*) - \max_{\mathbf{x} \in \mathcal{D}_{1:t}} f(\mathbf{x})$, where $f(\mathbf{x}^*)$ is the true optima of the objective function. We do not consider any preference based BO methods [24, 26] as baselines, because the preferences are provided directly on the objective function, as opposed to abstract properties that are not measured directly. The additional details of our experimental setup are provided in the supplementary material (Sect. 11).

5.1 Synthetic Experiments

We evaluate BOAP framework in the global optimization of synthetic benchmark functions [32]. The list of synthetic functions used are provided in Table 1.

Emulating Preferential Expert Inputs: As discussed in Sect. 3.1, we fit a rank GP using the expert preferences provided on designs based on their cognitive knowledge. In all our synthetic experiments we set $m = 2$, *i.e.*, we model two abstract properties $\{\omega_1, \omega_2\}$ for the considered synthetic function. We expect the expert to know the higher order abstract features of each design $\mathbf{x} \in \mathcal{X}$. We construct rank GPs by emulating the expert preferences based on such high level features of the given synthetic function. The possible set of high level features of the synthetic functions are mentioned in Table 1. We generate preference list P^{ω_i} for each high level feature of the designs by comparing its utility. We start with $p = \binom{t'}{2}$ preferences in P , that gets updated in every iteration of the optimization process. We construct rank GP surrogates $\{\mathcal{GP}_{\omega_1}, \mathcal{GP}_{\omega_2}\}$ using P^{ω_1} and P^{ω_2} .

For a given d -dimensional problem, we have considered $t' = d + 3$ initial observations and allocate $T = 10 \times d + 5$ budget. We repeat all our synthetic experiments 10 times with random initialization and report the average simple regret [12] (along with its standard error) as a function of iterations. The

convergence plots obtained for the optimization of synthetic functions after 10 runs are shown in Fig. 3. It is evident from the convergence results that our proposed BOAP method has outperformed the standard baselines by a huge margin, thereby proving its superiority. Further, it is also observed that BOAP-OA, a BOAP variant without the 2-arm bandit strategy and just the augmented GP (\mathcal{GP}_h), has a superior performance when compared to the baselines (BO-EI and BO-TS), thereby indicating the usefulness of expert inputs in significantly improving the performance of Bayesian optimization algorithm.

Table 1. Details of the synthetic optimization benchmark functions. Analytical forms are provided in the 2nd column and the last column depicts the high level features used by a simulated expert.

| Functions | $f(\mathbf{x})$ | High Level Features |
|---------------|--|--|
| Benchmark-1D | $\exp^{(2-\mathbf{x})^2} + \exp^{\frac{(6-\mathbf{x})^2}{10}} + \frac{1}{\mathbf{x}^2+1}$ | $\omega_1 = \exp^{(2-\mathbf{x})^2}, \omega_2 = \frac{1}{\mathbf{x}^2}$ |
| Rosenbrock-3D | $\sum_{i=1}^{d-1} [100 \times (x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$ | $\omega_1 = (x_3 - x_2^2)^2 + (x_2 - x_1^2)^2$ $\omega_2 = (x_2 - 1)^2 + (x_1 - 1)^2$ |
| Griewank-5D | $\sum_{i=1}^d \left[\frac{x_i^2}{4000} - \prod_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \right]$ | $\omega_1 = \sum_{i=1}^d x_i^2, \omega_2 = \prod_{i=1}^d \cos x_i$ |

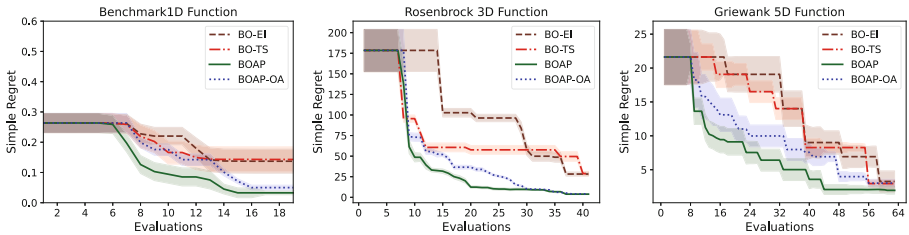


Fig. 3. Simple regret vs iterations for robustness experiments using synthetic multi-dimensional benchmark functions. We plot the average regret (along with its standard error) obtained after 10 random repeated runs.

To demonstrate the robustness of our approach we have conducted additional experiments by accounting for the inaccuracy or poor choices in expert preferential knowledge. Here, we show the robustness of our BOAP approach in two scenarios. First, we show the performance of our proposed approach when the higher order abstract properties are poorly selected. Second, we incorporate noise in the expert preferential feedback by flipping the expert preference between two inputs (designs) with a probability δ . We now discuss in detail the aforementioned two variations of our proposed method.

Inaccurate Abstract Properties (BOAP-IA). In the first variation, we assume that the expert poorly selects the human abstraction features. Table 2 depicts the synthetic functions considered and the corresponding (poorly chosen or uninformative) human abstraction features (ω_1 and ω_2). BOAP-IA uses such inaccurate human abstract features while augmenting the original input space.

Table 2. Selection of abstract (uninformative) features by a simulated human expert. The human abstraction (high level) features shown in the 3rd column are deliberately selected to be uninformative.

| Functions | $f(\mathbf{x})$ | Human Abstraction Features |
|---------------|--|--|
| Benchmark-1D | $\exp^{(2-\mathbf{x})^2} + \exp^{\frac{(6-\mathbf{x})^2}{10}} + \frac{1}{\mathbf{x}^2+1}$ | $\omega_1 = \sin \mathbf{x}, \omega_2 = \cos \mathbf{x}$ |
| Rosenbrock-3D | $\sum_{i=1}^{d-1} [100 \times (x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$ | $\omega_1 = \sin \mathbf{x}, \omega_2 = \cos \mathbf{x}$ |
| Griewank-5D | $\sum_{i=1}^d \left[\frac{x_i^2}{4000} - \prod_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \right]$ | $\omega_1 = \sin \mathbf{x}, \omega_2 = \mathbf{x}^3$ |

Noisy Expert Preferences (BOAP-NP). In the second variation, we account for the inaccurate expert preferential knowledge by introducing an error in human expert preferential feedback. To do this, we flip the preference ordering with a probability δ i.e., $P^{\omega, \delta} = \{(\mathbf{x}_i \succ \mathbf{x}_j) \mid \mathbf{x}_i, \mathbf{x}_j \in \mathbf{x}_{1:n}, \nu_{ij} \omega(\mathbf{x}_i) > \nu_{ij} \omega(\mathbf{x}_j)\}$, where ν_{ij} is drawn from a random distribution such that it is +1 with probability $1 - \delta$, -1 with probability δ . In this set of experiments we have set the probability $\delta = 0.3$.

We evaluate the performance by computing the simple regret after $10 \times d$ iterations. The empirical results for BOAP with inaccurate features (BOAP-IA) and BOAP framework with noisy preferences (BOAP-NP) are presented in Fig. 4. Although the expert preferential knowledge is noisy and inaccurate, it is significant from the results that our proposed BOAP framework outperforms the standard baselines. We believe that the superior performance of BOAP variants is due to the model selection based safeguard mechanism that uses 2-arm scheme to intelligently select the arm with the maximum predictive likelihood to suggest the next sample.

5.2 Real-World Experiments

We demonstrate the performance of BOAP in two real-world optimization use-cases in Lithium-ion battery manufacturing that are proven to be very complex and expensive in nature, thus providing a wide scope for the optimization. Further, battery scientists often reveal additional knowledge about the abstract properties in the battery design space and thus providing a rich playground for the evaluation of our framework. We refer to the supplementary material (Sect. 11.2) for the detailed experimental setup.

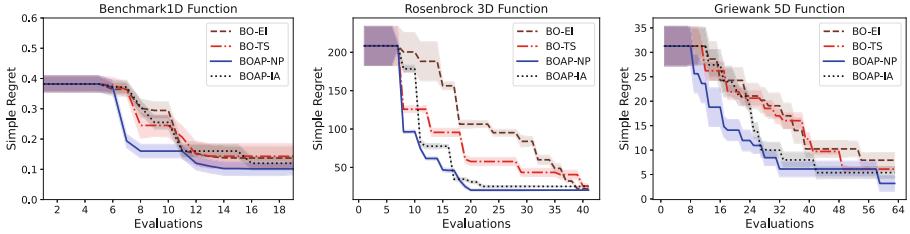


Fig. 4. Simple regret vs iterations for the synthetic multi-dimensional benchmark functions. We plot the average regret (along with its standard error) obtained after 10 random repeated runs.

Optimization of Electrode Calendering. In this experiment, we consider a case study on the calendering process proposed in [33]. The authors analyzed the effect of parameters such as calendering pressure (ε_{cal}), electrode porosity and electrode composition on the electrode properties such as electrolyte conductivity, tortuosity (both in solid phase (τ_{sol}) and liquid phase (τ_{liq})), Current Collector (CC), Active Surface (AS), etc. We define an optimization paradigm using the data grid published in [33].

We use our proposed BOAP framework to optimize the electrode calendering process by maximizing the *Active Surface* of electrodes by modeling two abstract properties: **(i)** Property 1 (ω_1): *Tortuosity in liquid phase* τ_{liq} , and **(ii)** Property 2 (ω_2): *Output Porosity* (OP). We simulate the expert pairwise preferential inputs $\{P^{\omega_{\tau_{\text{liq}}}}, P^{\omega_{\text{OP}}}\}$ by comparing the actual measurements reported in the dataset published in [33]. We consider 4 initial observations and maximize the active surface of the electrodes for 50 iterations. We compare the performance of our proposed BOAP framework by plotting the average simple regret (along with its standard error) after 10 repeated runs with random initialization. The convergence results obtained for the electrode optimization are shown in Fig. 5a.

Electrode Manufacturing Optimization. The best battery formulation and the optimal selection of process parameters is crucial for manufacturing long-life and energy-dense batteries. [34] analyzed the manufacturing of Lithium-ion graphite based electrodes and reported the process parameters in manufacturing a battery along with the output charge capacities of the battery measured after certain charge-discharge cycles. In our experiment, we use BOAP to optimally select the manufacturing process parameters to design a battery with maximum endurance *i.e.*, a battery that can retain the maximum charge after certain charge-discharge cycles. We consider *Anode Thickness* (AT) and *Active Mass* (AM) as abstract properties $\{\omega_{\text{AT}}, \omega_{\text{AM}}\}$ to maximize the battery endurance $E = \frac{D_{50}}{D_5}$, where D_{50} and D_5 are the discharge capacities of the cell at 50th and 5th cycle, respectively. We consider 4 initial observations and maximize the endurance of the cell for 50 iterations. We compare the performance by plotting

the average simple regret versus iterations after 10 random repeated runs. The convergence results obtained for maximizing the endurance is shown in Fig. 5b.

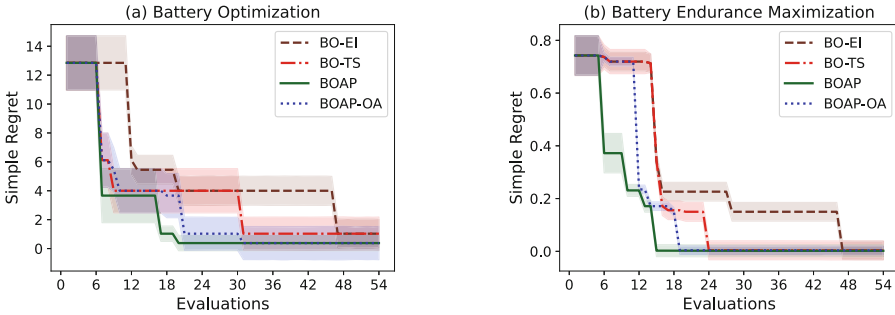


Fig. 5. Simple regret vs iterations for battery manufacturing optimization experiments: (a) Optimization of electrode calendaring process (b) Optimization of the battery endurance.

It is evident from Fig. 5 that BOAP is superior to the baselines due to its ability to model the abstract properties of the battery designs that can be beneficial in accelerating BO performance. Similar to the trends observed in the synthetic experiments, BOAP-OA with just the augmented inputs has outperformed the standard baselines (BO-EI and BO-TS), thereby proving again the benefits of expert inputs in boosting the optimization performance. The supplementary material along with the necessary implementation details and the code snippets are available at <https://github.com/mailtoarunkumarav/BOAP>.

6 Conclusion

We present a novel approach for human-AI collaborative BO for modeling the expert inputs on abstract properties to further improve the sample-efficiency of BO. Experts provide preferential inputs about the abstract and unmeasurable properties. We model such preferential inputs using rank GPs. We augment the inputs of a standard GP with the output of such auxiliary rank GPs to learn the underlying preferences in the instance space. We use a 2-arm strategy, a key safeguard that provides assurance to utilize only relevant and accurate expert preferential inputs in the modeling, thus overcoming any futile expert bias. We discuss the convergence of our proposed BOAP framework. The experimental results show the superiority of our proposed BOAP algorithm.

Acknowledgements. This research was partially supported by the Australian Government through the Australian Research Council’s Discovery Project funding scheme (project DP210102798). The views expressed herein are those of the authors and are not necessarily those of the Australian Government or Australian Research Council.

Disclosure of Interests. Prof. Svetha Venkatesh has received research grants from Australian Research Council’s Discovery Project funding scheme.

References

1. Martinez-Cantin, R.: BayesOpt: a Bayesian optimization library for nonlinear optimization, experimental design and bandits. *J. Mach. Learn. Res.* **15**(1), 3735–3739 (2014)
2. Greenhill, S., Rana, S., Gupta, S., Vellanki, P., Venkatesh, S.: Bayesian optimization for adaptive experimental design: a review. *IEEE access* **8**, 13937–13948 (2020)
3. Srinivas, N., Krause, A., Kakade, S.M., Seeger, M.W.: Information-theoretic regret bounds for Gaussian process optimization in the bandit setting. *IEEE Trans. Inf. Theory* **58**(5), 3250–3265 (2012)
4. Chowdhury, S.R., Gopalan, A.: On kernelized multi-armed bandits. In: Precup, D., Teh, Y.W. (eds.) *Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research*, vol. 70, pp. 844–853. PMLR, International Convention Centre, Sydney, Australia (Aug 2017)
5. Swersky, K.: *Improving Bayesian optimization for machine learning using expert priors*. University of Toronto (Canada) (2017)
6. Venkatesh, A.K.A., Rana, S., Li, C., Gupta, S., Shilton, A., Venkatesh, S.: Bayesian optimization for objective functions with varying smoothness. In: *Australasian Joint Conference on Artificial Intelligence*, pp. 460–472 (2019)
7. Li, C., et al.: Accelerating experimental design by incorporating experimenter hunches. In: *2018 IEEE International Conference on Data Mining (ICDM)*, pp. 257–266 (2018). <https://doi.org/10.1109/ICDM.2018.00041>
8. Hvarfner, C., Stoll, D., Souza, A., Lindauer, M., Hutter, F., Nardi, L.: π BO: Augmenting Acquisition Functions with User Beliefs for Bayesian Optimization. *arXiv preprint arXiv:2204.11051* (2022)
9. Venkatesh, A.K.A., Rana, S., Shilton, A., Venkatesh, S.: Human-AI Collaborative Bayesian optimization. In: *Advances in Neural Information Processing Systems* (2022)
10. Nguyen, Q.P., Tay, S., Low, B.K.H., Jaillet, P.: Top-k ranking Bayesian optimization. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 9135–9143 (2021)
11. Williams, C.K., Rasmussen, C.E.: *Gaussian Processes For Machine Learning*, vol. 2. MIT press Cambridge, MA (2006)
12. Brochu, E., Cora, V.M., De Freitas, N.: A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599* (2010)
13. Frazier, P.I.: A tutorial on Bayesian optimization. *arXiv preprint arXiv:1807.02811* (2018)
14. Kushner, H.J.: A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *J. Basic Eng.* **86**(1), 97–106 (03 1964). <https://doi.org/10.1115/1.3653121>
15. Shah, A., Wilson, A., Ghahramani, Z.: Student-t processes as alternatives to Gaussian processes. In: *Artificial intelligence and statistics*, pp. 877–885 (2014)
16. Zhang, Z., Si, X., Hu, C., Lei, Y.: Degradation data analysis and remaining useful life estimation: a review on Wiener-process-based methods. *Eur. J. Oper. Res.* **271**(3), 775–796 (2018)
17. Mockus, J., Tiesis, V., Zilinskas, A.: The application of Bayesian methods for seeking the extremum. In: *Towards Global Optimization*, vol. 2, pp. 117–129 (September 1978)

18. Thompson, W.R.: On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* **25**(3–4), 285–294 (1933)
19. Kahneman, D., Tversky, A.: Prospect theory: An analysis of decision under risk. In: *Handbook of the Fundamentals of Financial Decision Making: Part I*, pp. 99–127. World Scientific (2013)
20. Siroker, D., Koomen, P.: *A/B testing: The most powerful way to turn clicks into customers*. John Wiley & Sons (2015)
21. Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.): *The Adaptive Web: Methods and Strategies of Web Personalization*. Springer Berlin Heidelberg, Berlin, Heidelberg (2007)
22. Herbrich, R., Minka, T., Graepel, T.: TrueSkill: a Bayesian skill rating system. In: *Advances in Neural Information Processing Systems* **19** (2006)
23. Chu, W., Ghahramani, Z.: Preference learning with Gaussian processes. In: *Proceedings of the 22nd International Conference on Machine learning*, pp. 137–144 (2005)
24. González, J., Dai, Z., Damianou, A., Lawrence, N.D.: Preferential Bayesian optimization. In: *International Conference on Machine Learning*, pp. 1282–1291. PMLR (2017)
25. Mikkola, P., Todorović, M., Järvi, J., Rinke, P., Kaski, S.: Projective preferential Bayesian optimization. In: *International Conference on Machine Learning*, pp. 6884–6892. PMLR (2020)
26. Benavoli, A., Azzimonti, D., Piga, D.: Preferential Bayesian optimization with skew Gaussian processes. In: *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 1842–1850 (2021)
27. Astudillo, R., Frazier, P.: Multi-attribute Bayesian optimization with interactive preference learning. In: *International Conference on Artificial Intelligence and Statistics*, pp. 4496–4507. PMLR (2020)
28. Huang, D., Filstroff, L., Mikkola, P., Zheng, R., Kaski, S.: Bayesian optimization augmented with actively elicited expert knowledge. *arXiv preprint [arXiv:2208.08742](https://arxiv.org/abs/2208.08742)* (2022)
29. Thurstone, L.L.: A law of comparative judgment. In: *Scaling*, pp. 81–92. Routledge (2017)
30. Kandasamy, K., Krishnamurthy, A., Schneider, J., Póczos, B.: Parallelised Bayesian optimisation via thompson sampling. In: *International Conference on Artificial Intelligence and Statistics*, pp. 133–142 (2018)
31. Neal, R.M.: *Bayesian learning for neural networks*, vol. 118. Springer Science & Business Media (2012)
32. Surjanovic, S., Bingham, D.: *Virtual library of simulation experiments: Test Functions and Datasets* (2017). <http://www.sfu.ca/~ssurjano>. Accessed 10 Apr 2024
33. Duquesnoy, M., Lombardo, T., Chouchane, M., Primo, E.N., Franco, A.A.: Data-driven assessment of electrode calendaring process by combining experimental results, in silico mesostructures generation and machine learning. *J. Power Sources* **480**, 229103 (2020)
34. Drakopoulos, S.X., et al.: Formulation and manufacturing optimization of Lithium-ion graphite-based electrodes via machine learning. *Cell Reports Phys. Sci.* **2**(12), 100683 (2021)