Recent Advances in Bayesian Optimization

Xilu Wang, Yaochu Jin, Sebastian Schmitt, Markus Olhofer

2023

Preprint:

Recent Advances in Bayesian Optimization

XILU WANG and YAOCHU JIN, Faculty of Technology, Bielefeld University, Germany
SEBASTIAN SCHMITT and MARKUS OLHOFER, Honda Research Institute Europe GmbH, Germany

Bayesian optimization has emerged at the forefront of expensive black-box optimization due to its data efficiency. Recent years have witnessed a proliferation of studies on the development of new Bayesian optimization algorithms and their applications. Hence, this paper attempts to provide a comprehensive and updated survey of recent advances in Bayesian optimization that are mainly based on Gaussian processes and identify challenging open problems. We categorize the existing work on Bayesian optimization into nine main groups according to the motivations and focus of the proposed algorithms. For each category, we present the main advances with respect to the construction of surrogate models and adaptation of the acquisition functions. Finally, we discuss the open questions and suggest promising future research directions, in particular with regard to heterogeneity, privacy preservation, and fairness in distributed and federated optimization systems.

CCS Concepts: • **General and reference → Surveys and overviews**; • **Theory of computation → Bayesian analysis**; • **Mathematics of computing → Nonparametric statistics**.

Additional Key Words and Phrases: Bayesian optimization, Gaussian process, acquisition function

1 INTRODUCTION

Optimization problems are pervasive in scientific and industrial fields, such as artificial intelligence, data mining, bioinformatics, software engineering, scheduling, manufacturing, and economics. Among them, many applications require to optimize objective functions that are noisy and expensive to evaluate, or do not have closed-form expressions, let alone gradient information. For such problems, metaheuristics such as evolutionary algorithms that rely on function values only are very popular. However, these algorithms usually require a large number of function evaluations. By contrast, Bayesian optimization (BO) has emerged as a mainstream to tackle these difficulties due to its high data efficiency, thanks to its ability to incorporate prior beliefs about the problem to help guide the sampling of new data, and to achieve a good balance between exploration and exploitation in the search.

Consider the maximization of an unknown function f that is expensive to evaluate, which can be formulated as follows:

$$x^* = \arg\max_{x \in \mathcal{X}} f(x) \tag{1}$$

where $\mathcal{X}$ denotes the search/decision space of interest and x^* is the global maximum. In principle, BO constructs a probabilistic model (also known as a surrogate model) that defines a distribution over the objective function, and then subsequently refines this model once new data is sampled. Specifically, BO first specifies a prior distribution

This work was supported in part by an Alexander von Humboldt Professorship for Artificial Intelligence endowed to Y. Jin by the German Federal Ministry of Education and Research and in part by the Honda Research Institute Europe.

Authors' addresses: Xilu Wang; Yaochu Jin, Faculty of Technology, Bielefeld University, 33619 Bielefeld, Germany, {xilu.wang;yaochu.jin}@uni-bielefeld.de; Sebastian Schmitt; Markus Olhofer, Honda Research Institute Europe GmbH, Carl-Legien-Strasse 30, 63073 Offenbach/Main, Germany, {sebastian.schmitt;markus.olhofer}@honda-ri.de.

0360-0300/2023/1-ART $15.00
https://doi.org/10.1145/3582078

over the function, which represents our belief about the objective function. Then, conditioned on the observed data and the prior, the posterior can be calculated using the Bayes rule, which quantifies our updated belief about the unknown objective function. As a result, the next sample can be identified by leveraging the posterior. This is achieved by optimizing some auxiliary functions, called acquisition functions (AFs) in BO. The workflow of BO is presented in the Supplementary material.

The origin of BO can be dated back to the work by Harold Kushner [99], where Wiener processes were adopted for unconstrained one-dimensional optimization problems and the probability of improvement is maximized to select the next sample. Mockus [124] developed a new AF, called expectation of improvement (EI), which was further used in [219]. Bayesian optimization was made popular in engineering after Jones *et al.* [86] introduced Efficient Global Optimization (EGO). In EGO, a Kriging model, called Design and Analysis of Computer Experiments stochastic process model [157], is adopted to provide best linear unbiased predictions of the objective, which is achieved by minimizing the Mean Squared Error of the predictor [97]. In BO, by contrast, a Gaussian process (GP) is adopted as the surrogate model, which is fit by maximizing the likelihood. Hence, the original formulation of Kriging is different from the GP [30]. An introduction to Kriging can be found in the Supplementary material. More recently, various variants of Kriging have been developed [79, 181] by accounting for constraints and noise in the optimization. As a result, Kriging models in spatial statistics are equivalent to GPs in BO in some papers, therefore the two terms will be used interchangeably in the rest of this paper. While GPs are the most commonly used surrogate models in BO, various alternatives have been proposed, such as Bayesian neural networks [61, 70], Bayesian linear regression [9, 169], deep GPs [65], random forests [81], ensembles [62], and dropout deep neural networks [63]. The past decades have witnessed a rapid development of BO in many real-world problems, including materials design and discovery, sensor networks, financial industry, and experimental design. More recently, BO became popular in machine learning, including reinforcement learning [179], hyperparameter tuning [14], and neural architecture search [93].

1.1 Related Surveys

There are already a few comprehensive surveys and tutorials on methodological and practical aspects of BO, each with a specific focus. Sasena [158] gave a review of early work on Kriging and its extension to constrained optimization. A tutorial on BO with GPs was given in [21], focusing on extending BO to active user modeling in preference galleries and hierarchical control problems. Shahriari *et al.* [164] presented a comprehensive review of the fundamentals of Bayesian optimization, elaborating on the statistical modeling and popular AFs. In addition, Frazier [47] discussed some recent advances in Bayesian optimization, in particular in multi-fidelity optimization and constrained optimization. However, none of the above review papers provides a comprehensive coverage of abundant extensions of BO. Moreover, many new advances in BO have been published since [164]. Hence, an updated and comprehensive survey of this dynamic research field will be beneficial for researchers and practitioners.

1.2 Contributions and Organization

This paper starts with a brief introduction to the fundamentals of Bayesian optimization in Section 2, including GPs and commonly used AFs. Section 3 provides a comprehensive review of the state-of-the-art, where a taxonomy of existing work on BO is proposed to offer a clear structure of the large body of research reported in the literature, as illustrated in Fig. 1. In this taxonomy, we divide most existing BO algorithms into nine groups according to the nature of the optimization problems. For each group, we attempt to include representative and state-of-the-art algorithms and methodologies. Since there is no systematic empirical comparison of all these algorithms, this paper predominantly provides conceptual and qualitative comparisons. We further introduce a color-coding

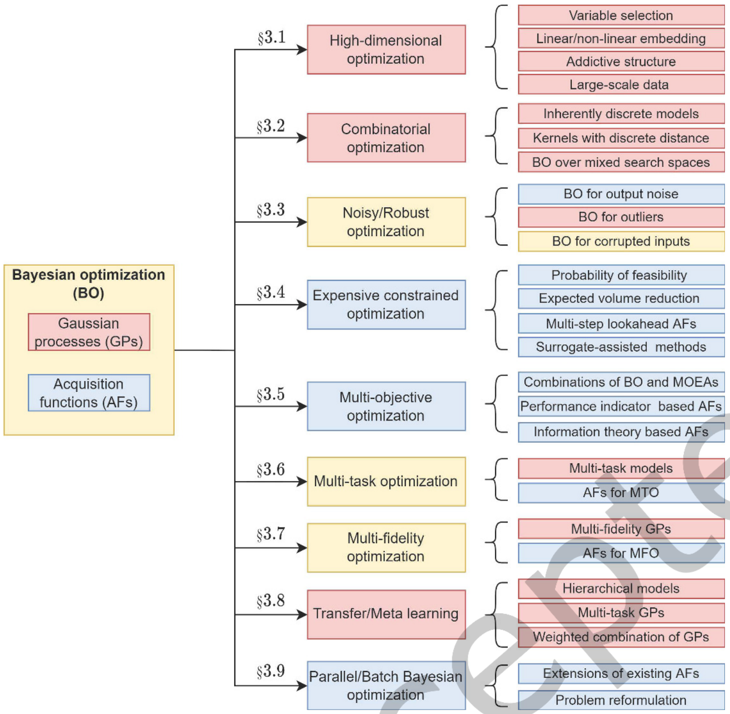
Bayesian optimization (BO)
Gaussian processes (GPs)
Acquisition functions (AFs)
§3.1
High-dimensional optimization
Variable selection
Linear/non-linear embedding
Addictive structure
Large-scale data
§3.2
Combinatorial optimization
Inherently discrete models
Kernels with discrete distance
BO over mixed search spaces
§3.3
Noisy/Robust optimization
BO for output noise
BO for outliers
BO for corrupted inputs
§3.4
Expensive constrained optimization
Probability of feasibility
Expected volume reduction
Multi-step lookahead AFs
Surrogate-assisted methods
§3.5
Multi-objective optimization
Combinations of BO and MOEAs
Performance indicator based AFs
Information theory based AFs
§3.6
Multi-task optimization
Multi-task models
AFs for MTO
§3.7
Multi-fidelity optimization
Multi-fidelity GPs
AFs for MFO
§3.8
Transfer/Meta learning
Hierarchical models
Multi-task GPs
Weighted combination of GPs
§3.9
Parallel/Batch Bayesian optimization
Extensions of existing AFs
Problem reformulation

Fig. 1. Taxonomy of Bayesian optimization algorithms. In the diagram, BO stands for Bayesian optimization, GP for Gaussian process, AF for acquisition function, MOEA for multi-objective evolutionary algorithm, MFO for multi-fidelity optimization, and MTO for multi-task optimization.

scheme to highlight the focuses of each group, where red, blue and yellow blocks indicate, respectively, a focus on AFs, surrogates, or both. Finally, this survey explores the challenges and a few emerging topics in BO.

2 FUNDAMENTALS OF BAYESIAN OPTIMIZATION

GPs and AFs are two main components of BO, which are introduced in the following.

2.1 Gaussian Process

GP is the most widely used probabilistic surrogate model for approximating the true objective function in BO. GP is characterized by a prior mean function $\mu(\cdot)$ and a covariance function $\kappa(\cdot,\cdot)$ [152]. Consider a finite collection of data pairs $\mathcal{D}_n = (\mathbf{X}, \mathbf{y})$ of the unknown function $y = f(\mathbf{X}) + \epsilon$ with noise $\epsilon \sim \mathcal{N}\left(0, \sigma_\epsilon^2\right)$, where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]^T$ is the input and $\mathbf{y} = [y_1, y_2, \cdots, y_n]^T$ is the output resulting from the true objective evaluations, and n is the number of samples. The GP model assumes that the observed data are drawn from a Gaussian distribution. Therefore, for a new data point $\mathbf{x}$, the joint distribution of the observed outputs $\mathbf{y}$ and the predicted output y are

$$\begin{bmatrix} \mathbf{y} \\ y \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(\mathbf{X},\mathbf{X}) + \sigma_\epsilon^2 \mathbf{I} & K(\mathbf{X},\mathbf{x}) \\ K(\mathbf{X},\mathbf{x})^T & \kappa(\mathbf{x},\mathbf{x}) \end{bmatrix}\right) \tag{2}$$

where T denotes matrix transposition, $K(\mathbf{X},\mathbf{X}) = [\kappa(\mathbf{x}_i, \mathbf{x}_j)]_{\mathbf{x_i},\mathbf{x_j} \in \mathbf{X}}$ denotes an $n \times n$ correlation matrix, and $K(X, \mathbf{x}) = [\kappa(\mathbf{x}_i, \mathbf{x})]_{\mathbf{x_i} \in \mathbf{X}}$ denotes a correlation vector evaluated at all pairs of training and test points. As described in [152], the conditional distribution $p(y \mid \mathbf{x}, \mathbf{X}, \mathbf{y}) \sim \mathcal{N}(\mu(\mathbf{x}), \sigma^2(\mathbf{x}))$ is then a multivariate Gaussian distribution, where the mean and variance of the predicted output y can be estimated as

$$\begin{aligned} \mu(\mathbf{x}) &= K(\mathbf{x}, \mathbf{X}) \, (K(\mathbf{X}, \mathbf{X}) + \sigma_\epsilon^2 \mathbf{I})^{-1} \mathbf{y} \\ \sigma^2(\mathbf{x}) &= (\mathbf{x}, \mathbf{x}) - K(\mathbf{X}, \mathbf{x})^T \, (K(\mathbf{X}, \mathbf{X}) + \sigma_\epsilon^2 \mathbf{I})^{-1} K(\mathbf{X}, \mathbf{x}) \, . \end{aligned} \tag{3}$$

Commonly used kernel functions are the squared exponential (Gaussian) kernel and the Matérn kernel [47], where hyperparameters, such as length scale, signal variance, and noise variance need to be specified. Take the squared exponential kernel as an example, let $\mathbf{x}$ and $\mathbf{x}'$ denote the inputs of two points,

$$k_{\mathrm{SE}}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{(\mathbf{x} - \mathbf{x}')^2}{2\ell^2} \right) \tag{4}$$

where ℓ defines the length-scale parameter, and σ_f is the single variance. In general, the free parameters, i.e., ℓ, σ_f, and σ_ϵ are called hyperparameters, denoted as $\theta = (\ell, \sigma_f, \sigma_\epsilon)$.

Typically, the optimal hyperparameters are inferred by maximizing the log marginal likelihood,

$$\log p(\mathbf{y} \mid \mathbf{X}, \theta) = -\frac{1}{2} \mathbf{y}^T \mathbf{K}_y^{-1} \mathbf{y} - \frac{1}{2} \log \; \mathbf{K}_y \; - \frac{n}{2} \log 2\pi \tag{5}$$

where $\mathbf{K}_y = K(\mathbf{X}, \mathbf{X}) + \sigma_\epsilon^2 \mathbf{I}$.

2.2 Acquisition Function

AFs are the utility functions that guide the search to reach the optimum of the objective function by identifying where to sample next, which is crucial in BO. The guiding principle behind AFs is to strike a balance between exploration and exploitation according to the uncertainty and optimality of the response surface, which is achieved by querying samples from both known high-fitness-value regions exploitation) and regions that have not been sufficiently explored so far (exploration). In the following, we briefly revisit the commonly used AFs and an illustration for some commonly used AFs can be found in the Supplementary material.

Without loss of generality, we consider a maximization problem. Let f^* denote the optimum obtained so far, and $\Phi(\cdot)$ and $\phi(\cdot)$ denote the normal cumulative distribution function (CDF), and probability density function (PDF) of the standard normal random variable, respectively. The earliest AF is to maximize the *probability of improvement* (PI) [99] over the current best value f^*, formulated as

$$\mathrm{PI}(\mathbf{x}; \mathcal{D}_n) = P\left(f(\mathbf{x}) \geq f^*\right) = \Phi\left(\frac{\mu(\mathbf{x}) - f^*}{\sigma(\mathbf{x})} \right) , \tag{6}$$

where P is the probability for finding a better objective function value at position $\mathbf{x}$ than the currently best value f^*. Alternatively, *expected improvement* [124] calculates the expected improvement with respect to f^*,

$$\begin{aligned} \mathrm{EI}(\mathbf{x}; \mathcal{D}_n) &= \mathbb{E} \; \max\left(0, f(\mathbf{x}) - f^*\right) \\ &= \left(\mu(\mathbf{x}) - f^*\right) \Phi\left(\frac{\mu(\mathbf{x}) - f^*}{\sigma(\mathbf{x})} \right) + \sigma(\mathbf{x}) \phi\left(\frac{\mu(\mathbf{x}) - f^*}{\sigma(\mathbf{x})} \right) , \end{aligned} \tag{7}$$

where $\mathbb{E}$ denotes the expectation value. Interested readers are referred to [207] for a comprehensive review of many variants of EI. Note, however, that EI tends to explore around the initial best point before the algorithm begins to search more globally, as only points that are close to the current best point have high EI values.

An idea closely related to EI is *Knowledge Gradient* (KG) [48], maximizing the expected incremental value of a measurement; however, it does not depend on the optimum obtained so far. Let μ_n denote the mean of the posterior distribution after n samples, and a new posterior distribution with posterior mean μ_{n+1} will be generated if we take one more sample. Hence, the KG is formulated as

$$\mathrm{KG}(\mathbf{x};\mathcal{D}_n) = \mathbb{E}_n \quad \max(\mu_{n+1}) - \max(\mu_n) \tag{8}$$

where $\mathbb{E}_n[\cdot] := \mathbb{E}[\cdot \mid \mathbf{X}, \mathbf{y}]$ indicates the conditional expectation with respect to what is known after the first n measurements.

The confidence bound criteria, *upper confidence bound* (UCB) for maximization problems and *lower confidence bound* (LCB) for minimization problems, are designed to achieve optimal regret in the multi-armed bandit community by combining the uncertainty and the expected reward [173]. The UCB is calculated as

$$\mathrm{UCB}(\mathbf{x};\mathcal{D}_n) = \mu(\mathbf{x}) + \beta\sigma(\mathbf{x}), \tag{9}$$

where $\beta > 0$ is a parameter to navigate the exploitation-exploration trade-off (LCB has a minus sign in front of the β term). Another promising AF for multi-armed bandit problems is *Thompson sampling* (TS) [3]. TS randomly draws each arm sampled from the posterior distribution, and then plays the arm with the highest simulated reward. A more recent development is the entropy-based AFs motivated by information theory, which can be further divided into input-entropy-based and output-entropy-based AFs. The former maximizes information about the location $\mathbf{x}^*$ of the global optimum where the information about $\mathbf{x}^*$ is measured by the negative differential entropy of the probability of the location of the global optimum, $p(\mathbf{x}^* \mid \mathcal{D}_n)$ [68, 73]. Hennig and Schuler [68] proposed *entropy search* (ES) using mutual information $I(\{\mathbf{x}, y\}; \mathbf{x}^* \mid \mathcal{D}_n)$,

$$\begin{aligned} \mathrm{ES} &= I(\{\mathbf{x}, y\}; \mathbf{x}^* \mid \mathcal{D}_n) \\ &= \mathrm{H} \;\; p(\mathbf{x}^* \mid \mathcal{D}_n) \;\; - \mathbb{E}_{p(y \mid \mathcal{D}_\mathbf{n}, \mathbf{x})} \;\; \mathrm{H} \;\; p(\mathbf{x}^* \mid \mathcal{D}_n \cup \{(\mathbf{x}, y)\}) \quad , \end{aligned} \tag{10}$$

where $\mathrm{H}[p(\mathbf{x})] = -\int p(\mathbf{x}) \log p(\mathbf{x}) d\mathbf{x}$ denotes the differential entropy and $\mathbb{E}_p[\cdot]$ denotes the expectation over a probability distribution p. However, the calculation in Eq. (10) is computationally intractable. To resolve this problem, Lobato *et al.* introduced *predictive entropy search* (PES) by equivalently rewriting Eq. (10) as

$$\mathrm{PES} = \mathrm{H} \;\; p(y \mid \mathcal{D}_n, \mathbf{x}) \;\; - \mathbb{E}_{p(\mathbf{x}^* \mid \mathcal{D}_\mathbf{n})} \;\; \mathrm{H} \;\; p(y \mid \mathcal{D}_n, \mathbf{x}, \mathbf{x}^*) \quad . \tag{11}$$

Compared with the previous formulation, PES is based on the entropy of predictive distributions, which is analytic or can be easily approximated. Following the same information-theoretic idea, output-entropy-based AFs maximize the reduction of the information about the maximum function value y^*, the mutual information $I(\{\mathbf{x}, y\}; y^* \mid \mathcal{D}_n)$ instead [192]. The *max-value entropy search* (MES) is formulated as

$$\begin{aligned} \mathrm{MES} &= I(\{\mathbf{x}, y\}; y^* \mid \mathcal{D}_n) \\ &= \mathrm{H}(p(y \mid \mathcal{D}_n, \mathbf{x})) - \mathbb{E}_{p(y^* \mid \mathcal{D}_\mathbf{n})} \;\; \mathrm{H}(p(y \mid \mathcal{D}_n, \mathbf{x}, y^*)) \quad . \end{aligned} \tag{12}$$

Intuitively, MES is computationally much simpler than ES and PES as MES uses one-dimensional $p(y^* \mid \mathcal{D}_n)$ while ES and PES estimate the expensive and multidimensional $p(\mathbf{x}^* \mid \mathcal{D}_n)$. Empirical results have demonstrated that MES performs at least as good as ES and PES [192].

Note that the above mentioned AFs are all designed for single-objective optimization, and therefore, many recent efforts have been dedicated to developing new AFs to account for a diverse and wide range of applications.

3 RECENT ADVANCES IN BAYESIAN OPTIMIZATION

In this section, we provide an overview of the state-of-the-art BO algorithms, focusing on the most important research advances. In the following, we categorize and discuss the existing work according to the characteristics of the optimization problems to provide a clear picture of the abundant literature.

3.1 High-dimensional optimization

High-dimensional black-box optimization problems are extremely challenging yet commonly seen in many applications [129, 190]. Note that the number of dimensions in BO may vary from dozens to thousands or even one billion [191]. Despite successful applications of BO to low-dimensional expensive and black-box optimization problems, BO is known to perform poorly when the dimension of the search space is larger than 10–20 [94, 105]. Hence, its extension to high-dimensional problems remains a critical open challenge. Specifically, the following major difficulties can be identified for BO of high-dimensional problems. 1) Nonparametric regression, such as GPs, is inherently difficult as the search space grows exponentially with the dimension. First, it becomes harder to learn a model in a high-dimensional space with the commonly used distance-based kernel functions, as the search spaces grow considerably faster than affordable sampling budgets. Second, the number of hyperparameters generally increases along with the input dimension, as a consequence, the training of the model becomes increasingly hard. 2) Generally, AFs are multi-modal problems, with a large mostly flat surface [150]. Hence, the optimization of AFs is non-trivial, in particular for high-dimensional problems and when the number of samples is limited. Note that the above problem is related to, but distinct from the scalability of GPs. To construct a reliable GP in higher dimensional space, more observed data may be required, which results in a challenge of scalability for the GP due to its cubic complexity to the data size. Although scalable GPs have been extensively studied in recent years to accommodate many observations [16, 112], these methods focus on the scenario where there exist a large amount of data while the dimension remains to be small or medium. Moreover, even if one can fit a GP for high-dimensional problems, one would still face the difficulty of the optimization of AFs, because AFs are typically multi-modal problems and require much more evaluations of the surrogate model to be optimized in high dimensions compared to low dimensions. Therefore, we are interested in scalable BO algorithms for tackling high dimensionality, rather than constructing high-dimensional GPs only.

Most existing BO algorithms for high-dimensional problems make two structural assumptions with few exceptions: 1) the high-dimensional objective function has a low active/effective dimensional subspace, which motivates the development of variable selection and embedding-based methods; 2) the original objective function can be a sum of several low-dimensional functions, which gives rise to additive structure based methods. Addressing high-dimensional BO with a large amount data generally involves alternative models, local modeling, and batch selection in a parallel manner. In the following, we will discuss in detail existing work handling high-dimensional optimization problems.

3.1.1 Variable selection.

To alleviate the curse of dimensionality, a straightforward idea is to adopt a dimension reduction technique. To achieve this, an important assumption often made is that the original objective function varies only within a low-dimensional subspace, called active/effective subspace [23]. To identify the most contributing input variables, some sensitivity analysis techniques that evaluate the relative importance of each variable with respect to a quantity of interest have been exploited [171]. In [23] two strategies, the finite difference sequential likelihood ratio test and the GP sequential likelihood ratio test, are proposed to screen the most contributing variables. Another commonly used quantity is the values of the correlation lengths of automatic relevance determination covariances [196]. The basic idea is that the larger the length scale value, the less important the corresponding variable.

3.1.2 Linear/non-linear embedding.

Instead of removing the inactive variables to reduce the dimension, more recent developments exploit the active dimensionality of the objective function by defining a latent space based on a linear or non-linear embedding. For example, Wang *et al.* [194] noted that given any $\mathbf{x} \in \mathbb{R}^D$ and a random matrix $\mathbf{A} \in \mathbb{R}^{D\times d}$, at a probability of 1, there is a point $\mathbf{y} \in \mathbb{R}^d$ such that $f(\mathbf{x}) = f(\mathbf{A}\mathbf{y})$. This observation allows us to perform BO in a low-dimensional space to optimize the original high-dimensional function. Hence, an algorithm, called BO with random embedding (REMBO), is proposed. Recently, several variants of REMBO have been reported [38, 105, 129]. Apart from the success in the random embedding methods, many algorithms have been proposed to learn the intrinsic effective subspaces, such as unsupervised learning based on variational autoencoders (VAE) [6], supervised learning [208], and semi-supervised learning [166]. Most of the above-mentioned methods based on the structural assumption use linear projections to scale BO to high dimensions. Recently, a few advanced techniques have been developed to further investigate the structure of the search space by using non-linear embeddings [125]. Compared with linear embeddings, non-linear embedding techniques, also known as geometry-aware BO [134], can be considerably more expressive and flexible. However, these methods require even more data to learn the embedding and assume that the search space is not Euclidean but various manifolds, such as Riemannian manifold [83].

3.1.3 Addictive structure.

The low active dimensionality assumption behind the aforementioned methods is too restrictive as all the input variables may contribute to the objective function. Hence, another salient structure assumption, called addictive structure, has been explored in the context of high-dimensional BO. The addictive structure has been used in addictive GPs [39]. An algorithm called Add-GP-UCB was proposed in [94], assuming that the objective function $f(\mathbf{x}) : \mathcal{X} \to \mathbb{R}$ with input space $\mathcal{X} = [0, 1]^D$ is a sum of functions of small, disjoint groups of dimensions,

$$f(\mathbf{x}) = f^{(1)} \;\; \mathbf{x}^{(1)} \;\; + f^{(2)} \;\; \mathbf{x}^{(2)} \;\; + \cdots + f^{(M)} \;\; \mathbf{x}^{(M)} \tag{13}$$

where $\mathbf{x}^{(j)} \in \mathcal{X}^{(j)} = [0, 1]^{d_\mathbf{j}}$ are disjoint subsets of variables. Instead of directly using addictive kernels, a set of latent decompositions of the feature space is generated randomly and the one with the highest GP marginal likelihood is chosen, with each kernel operating on subsets of the input dimensions. Markov Chain Monte Carlo (MCMC) [50], Gibbs sampling [193] and Thompson sampling [192] were also introduced to more effectively learn the addictive structure. Another major issue concerning Add-GP-UCB is the restriction of disjoint subsets of input dimensions, which have been lifted in subsequent work [108, 153]. Li *et al.* generalized the two structure assumptions, i.e., the low active assumption and the addictive structure assumption, by introducing a projected-addictive assumption. In [76, 153], overlapping groups are allowed by representing the addictive decomposition via a dependency graph or a sparse factor graph.

3.1.4 Large-scale data in high-dimensional Bayesian Optimization.

While there have been ample studies on BO to account for problems with large-scale observations and high-dimensional input spaces, very few have considered high-dimensional problems with a large amount of training data. This optimization scenario is indispensable as more data is required for constructing surrogates in high-dimensional spaces. Earlier research has shed some light on the potential advantages of replacing GPs with more scalable and flexible machine learning models. A natural choice is Bayesian neural networks due to their desirable flexibility and characterization of uncertainty [172]. Guo *et al.* [63] developed an efficient dropout neural network (EDN) to replace GPs in high-dimensional multi/many-objective optimization. The core idea in EDN is that the dropout is executed during both the training and prediction processes, so that EDN is able to estimate the uncertainty for its prediction. Alternatively, random forests [81] and the quadrature Fourier feature approximation [127] have been adopted to replace GPs to address large-scale high-dimensional problems. More recently, a few methods have been proposed that resort to local modeling and batch selection in a parallel manner to scale BO to problems with large-scale observations and high-dimensional input spaces. Wang *et al.* [190] proposed ensemble BO to alleviate the difficulties of constructing GPs and optimizing AFs for high-dimensional problems. Ensemble BO firstly learns local models on partitions of the input space and subsequently leverages the batch selection of new queries in each partition. Similarly, an MOEA with a heterogeneous ensemble model as a surrogate was proposed [62], in which each member is trained by different input features generated by feature selection or feature extraction. The trust region method is adopted to design a local probabilistic approach (namely TuRBO) for handling large-scale data in high-dimensional spaces [43]. However, the trust regions in TuRBO are learned independently without sharing data, which may be inefficient for expensive problems.

3.1.5 Discussions.

The two structural assumptions benefit the GP modeling in high-dimensional spaces, but may be violated in real-world applications, where the objective function or the search space is not decomposable. The remaining open questions include how to effectively learn the low-dimensional latent space. Recently, the presence of high-dimensional combinatorial optimization and graph structure objective functions pose challenges for BO, which deserves further investigation. Moreover, while most research work consider high-dimensional search spaces, investigation of high-dimensional multi-output BO still lacks.

3.2 Combinatorial optimization

The optimization of black-box functions over combinatorial spaces, e.g., integer, sets, sequences, categorical, or graph structured input variables, is ubiquitous and yet challenging task in real-world applications. Without loss of generality, suppose there is an expensive black-box objective function $f : \mathcal{H} \rightarrow \mathbb{R}$. The goal of combinatorial optimization is:

$$\mathbf{h}^* = \arg\max f(\mathbf{h}) \tag{14}$$

where $\mathcal{H}$ denotes the search space. For problems over a hybrid search space, $\mathcal{H} = [\mathcal{C}, \mathcal{X}]$, $\mathcal{C}$ and $\mathcal{X}$ denote the discrete and continuous search space, respectively. Specifically, discrete variables can be divided into ordinal and nominal (or quantitative and qualitative) variables according to whether a relation of order between the possible values of a given variable can be defined [139]. For example, categorical variables refer to an unordered set. BO has emerged as a well-established paradigm for handling costly-to-evaluate black-box problems. However, most Gaussian process-based BO algorithms explicitly assume a continuous space, incurring poor scalability to combinatorial domains. This can mainly be attributed to the difficulty in defining kernels and distance measures over combinatorial spaces to account for complex interactions between variables. Note that gradient-based methods for optimizing AFs are not directly applicable in the presence of discrete variables. Moreover, BO suffers seriously from the fact that the number of possible solutions grows exponentially with the parameters in the combinatorial domain (known as combinatorial explosion). Consequently, there are two major challenges for combinatorial BO. One is the construction of effective surrogate models over the combinatorial space, and the other is the effective search in the combinatorial domain for the next structure for evaluation according to the AF. A straightforward way is to construct GPs and optimize AFs by treating discrete variables as continuous, and then the closest integer for the identified next sample point with real values is obtained via a one-hot encoding strategy [52]. Clearly, this approach ignores the nature of the search space and may repeatedly select the same new samples, which deteriorates the efficiency of BO. Alternatively, many studies borrowed the elegance of VAEs to map high-dimensional, discrete inputs onto a lower dimensional continuous space [57]. In the context of BO, much effort has been dedicated to handling expensive combinatorial optimization problems by introducing surrogate models for combinatorial spaces.

3.2.1 Inherently discrete models.

To sidestep the difficulties encountered in the GP-based BO, some inherently discrete models (e.g. neural networks [176] and random forests) are employed as surrogate models, among which tree-based models are the most widely used ones. For example, random forests have been applied to the combinatorial BO in [80]. Unfortunately, this approach suffers from performing undesirable extrapolation. Hence, a tree-structured Parzen estimator model has been used to replace the GPs in [14], which, however, requires a large number of training data. An alternative idea is to use continuous surrogate models that guarantee integer-valued optima, which motivates a method called IDONE [17] using a piece-wise linear surrogate model. To improve the search efficiency of the AF in combinatorial optimization, search control knowledge is introduced to branch-and-bound search [37]. In addition, an algorithm called BOCS is proposed to alleviate the combinatorial explosion of the combinatorial space [9].

3.2.2 **Kernels with discrete distance measures.** Another popular avenue for combinatorial BO is to modify the distance measure in the kernel calculation of Gaussian processes, so that the similarity in the combinatorial space can be properly captured. For example, the Hamming distance is widely used to measure the similarity between discrete variables, and an evolutionary algorithm is generally adopted to optimize the AF [80]. More recently, graph presentations of combinatorial spaces has emerged at the forefront, contributing to graph kernels in GPs. Oh *et al.* [135] proposed COMBO, which constructs a combinatorial graph over the combinatorial search space, in which the shortest path between two vertices in the graph is equivalent to the Hamming distance. Subsequently, graph Fourier transforms are utilized to derive the diffusion kernel on the graph. To circumvent the computational bottleneck of COMBO, the structure of the graph representation is further studied and a small set of features is extracted [36]. Note that graph-based combinatorial BO has been widely applied to neural architecture search [93, 155].

3.2.3 **Bayesian optimization over mixed search spaces.** Very few studies have considered mixed-variable combinatorial problems, where the input variables involve both continuous and discrete ones, such as integers and categorical inputs. The kernels with new distance measures over discrete spaces have shed light on addressing combinatorial optimization problems. Hence, some attempts have been made for combinatorial BO in a similar fashion, i.e., combining kernels defined over different input variables [154]. Interestingly, Pelamatti *et al.* [138] used a product of kernels defined over different domains to address constrained mixed-variable problems. Following this, similar kernels are defined to address mixed-variable problems with varying-size search space [139]. While replacing the GPs in the framework of Bayesian optimization is a possible approach in the mixed-variable setting [17], the bandit approaches have been integrated with BO by treating each variable as a bandit [130].

3.2.4 **Discussions.** While most combinatorial BO methods focus on the construction of surrogate models, the combinatorial explosion problem remains challenging. The computational bottleneck and scalability challenges require new research ideas and deserve to be further investigated. Moreover, due to the constraints involved in combinatorial optimization, it is increasingly attractive to select new queries satisfying the constraints.

3.3 Noisy and robust optimization

Two assumptions about the noise in the data are made for constructing the GP in BO [120]. First, the measurement of the input points is noise-free. Second, noise in observations is often assumed to follow a constant-variance normal distribution, called homoscedastic Gaussian white noise. However, neither of these assumptions may hold in practice, rendering poor optimization performance. Hence, BO approaches accounting for noisy observations, outliers, and input-dependent noise have been developed.

3.3.1 **Bayesian optimization for output noise.** For an optimization with noisy output, the objective function can be described by $f : \mathcal{X} \to \mathbb{R}$ resulting from noisy observations $y = f(\mathbf{x}) + \epsilon$, where ϵ is addictive/output noise. Most BO approaches for problems in the presence of output noise employ the standard GP as the surrogate model and focus on designing new AFs [145]. Firstly, the extension of the noise-free EI (Eq. 7) to noisy observations has been studied extensively [207]. One major issue is that the current best objective value $f(\mathbf{x}^*)$ is not exactly known. A direct approach is to replace $f(\mathbf{x}^*)$ by some sensible values, which is called expected improvement with "plug-in" [145]. Huang *et al.* [79] developed an augmented EI by replacing the current best objective value and subsequently added a penalty term to the standard EI. Alternatively, the β-quantile given by the GP surrogate is used as a reference in [144]. In that work, an improvement based on the decrease of the lowest of the β-quantile is further defined, yielding the expected quantile improvement (EQI) that is able to account for heterogeneous noise. Similar to EQI, the improvement is defined by the KG policy, and an approximate knowledge gradient (AKG) is introduced [161]. Fundamentally, AKG is an EI based on the knowledge improvement; however, the evaluation of AKG is computationally intensive. Another class of AFs that naturally handles output noise is information-based AFs, such as the PES [73] and Thompson sampling algorithm [92].

A reinterpolation method was also proposed to handle output noise [46], where a Kriging regression is constructed using noisy observations. Then, the sampled points with the predictions provided by the Kriging are adopted to build an interpolating Kriging, which is called the reinterpolation, enabling the standard EI to select new samples.

3.3.2 Bayesian Optimization for outliers.

Besides the above mentioned measurement/output noise, the observations are often contaminated with outliers/extreme observations in real experiments due to irregular and isolated disturbances, instrument failures, or potential human errors. As pointed out in O'Hagan [136], the standard GP model that adopts Gaussian distributions as both the prior and the likelihood is sensitive to extreme observations. Another reason is that GP is nonparametric and interpolant, and therefore it will (in the classical settings with small variance noise) go through the outlier data.

Typically, BO adopts robust GPs that are insensitive to the presence of outliers to account for outliers. Mathematically, the main idea behind robust GP models is to use an appropriate noise model with a heavier tail, instead of assuming normal noise, to account for the outlying data [1]. The most commonly used noise model is Student-t distribution [118, 182]. However, using the Student-t likelihood will not allow a closed form of inference of the posterior distribution, therefore, some techniques of approximate inference are required. For example, the Laplace approximation [182] is used for approximate inference. More recently, Martinez-Cantin [118] proposed an outlier-handling algorithm by combining a robust GP with Student-t likelihood with outlier diagnostics to classify data points as outliers or inliers. Thus, the outliers can be removed and a standard GP can be performed, resulting in a more efficient robust method with a better convergence.

3.3.3 Bayesian optimization for corrupted inputs.

The input-dependent noise was first considered in modeling GP [55], where heteroscedastic noise was introduced by allowing the noise variance to be a function of input instead of a constant. Hence, the noise variance is considered as a random variable and an independent GP is used to model the logarithms of the noise level. The inference in heteroscedastic GP regression is challenging, since, unlike in the homoscedastic case, the predictive density and marginal likelihood are no longer analytically tractable. The MCMC method can be used to approximate the posterior noise variance, which is, however, time-consuming. Suggested alternative approximations include variational inference [102], Laplace approximation [182] and expectation propagation [1].

The above mentioned methods handle datasets with input noise by holding the input measurements as deterministic and changing the corresponding output variance to compensate. McHutchon and Rasmussen [120] pointed out that the effect of the input-dependent noise is related to the gradient of the function mapping input to output. Therefore, a noisy input GP (NIGP) was developed, where the input noise is transferred to output based on a first order Taylor expansion of the posterior. Specifically, NIGP adopts a local linearization of the function, and uses it to propagate uncertainty from the inputs to the output of the GP [120]. The intuition behind the above ideas is to propagate the input noise to the output space, which may, however, result in unnecessary exploration. Nogueira *et al.* [133] addressed this issue by considering input noise in EI, so that the input noise can be propagated through all the models and the function queries. More precisely, an unscented expected improvement and an unscented optimal incumbent are defined using the unscented transformation (UT). UT first deterministically chooses a set of samples from the original distribution. Then, a nonlinear function is applied to each sample to yield transformed points. Hence, the mean and covariance of the transformed distribution can be formed according to the weighted combination of the transformed points.

A closely related term to input-dependent noise is input/query uncertainty [13]. That is, the estimation of the actual query location is also subject to uncertainty, such as environmental variables [119] or noise-corrupted inputs. When extending BO to problems with input uncertainty, two classical problem formulations, a probabilistic robust optimization and worst-case robust optimization, from a probabilistic and deterministic point of view have been adopted. In probabilistic robust optimization, a distribution of the input or environmental variables is assumed. Hence, a prior is placed on the input space in order to account for localization noise, and performance is assessed by the expected value of some robustness measurement. A representative work by Bland and Nair [13] introduces noise-corrupted inputs, namely uncertainty, within the framework of Bayesian optimization. In this case, a robust optimization problem is formulated as a constrained problem by integrating an unknown function with respect to the input distributions. Hence, the noise factors can be integrated out and an AF similar to the constrained EI is introduced to select new queries entirely in the decision space. By contrast, the worst-case robust objective aims to search for a solution that is robust to the worst possible realization of the uncertain parameter, which is formulated as a min-max optimization problem,

$$\max_{\mathbf{x}} \min_{\mathbf{c} \in U} f(\mathbf{x}, \mathbf{c}), \tag{15}$$

where $\mathbf{x}$ denotes the decision vector, $\mathbf{c} \in U$ denotes uncertainties, where U is the uncertainty set. Marzat [119] uses a relaxation procedure to explore the use of EGO for worst-case robust optimization, so that the design variables and the uncertainty variables can be optimized iteratively. However, such a strategy is inefficient as the previous observations are not reused. Ur Rehman *et al.* [180] proposed a modified EI using a new expected improvement.

3.3.4 Discussions. New AFs are designed for addictive output noise, while enhancements of GPs based on Student-t distribution are developed to accommodate outliers. More recently, more complex problem settings with new robustness requirements in realistic scenario have attracted increased attention. For example, how to address adversarial corruptions [18] is one of the promising research directions. Moreover, robustness in batch optimization and bandit optimization [18] will be of paramount importance.

3.4 Expensive constrained optimization

Many optimization problems are subject to various types of constraints, and the evaluation of both the objective function and the constraints can be computationally intensive or financially expensive, known as expensive constrained optimization problems (ECOPs). Without loss of generality, an ECOP can be formulated as

$$\begin{array}{ll} \min_{\mathbf{x}} & \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x})) \\ \text{s.t.} & c_j(\mathbf{x}) \geq a_j, j = 1, 2, \ldots, q \\ & h_i(\mathbf{x}) = b_i, i = 1, 2, \ldots, r \\ & \mathbf{x} \in X \end{array} \tag{16}$$

where x $= (x_1, x_2, \ldots, x_d)$ is the decision vector with d decision variables, X denotes the decision space, $c_j(\mathbf{x})$ and $h_i(\mathbf{x})$ denote inequality and equality constraints, respectively. Since we consider both single-objective and multi-objective problems, the objective vector f consists of m objectives and $m = 1, 2, \cdots, N$. BO for ECOPs can be roughly classified into two groups. 1) With the help of GPs, new AFs are proposed to account for the constraints within the framework of BO, known as constrained BO (CBO). Recently, CBO has become popular, especially for addressing single-objective constrained problems. According to the different AFs in CBO, we classify various CBO algorithms into three sub-categories: probability of feasibility based, expected volume reduction based, and multi-step look-ahead methods. 2) To circumvent the computational burden encountered in ECOPs, BO is adopted in existing constraint-handling methods, typically, evolutionary algorithms. We refer to these as surrogate-assisted constraint-handling methods. In the following, each group is introduced and discussed.

3.4.1 Probability of feasibility. The combination of the existing AFs with constraint feasibility indicators, such as probability of feasibility, offers a principled approach to constrained optimization. The most representative work is the extension of the well-established EI, known as EI with constraints (EIC) [8]. One of the previous EIC methods, called constrained EI (cEI) or constraint-weighted EI, aims to maximize the expected feasible improvement over the current best feasible observation. Typically, cEI multiplies the EI and the constrained satisfaction probabilities, formulated as follows:

$$\mathrm{cEI}(\mathbf{x}) = EI(\mathbf{x}) \prod_{j=1} \Pr \ \hat{c}_j(\mathbf{x}) \leq a_j \tag{17}$$

where each constraint is assumed to be independent, all expensive-to-evaluate functions are approximated by independent GPs, and $\hat{c}_j$ denotes the model prediction for the j-th constraint. Interestingly, similar ideas have been discussed in [160] and revisited in [51]. As indicated in Equation (17), cEI faces several issues. First, the current best observation is required, which is untenable in some applications, such as noisy experiments. Hence, a recent work by Letham *et al.* [106] directly extends cEI to noisy observations with greedy batch optimization. Second, cEI can be brittle for highly constrained problems, because the product of feasibility probabilities approaches zeros near the feasibility border where the optimum is located, resulting in very small values of cEI in the interesting regions [8].

3.4.2 Expected volume reduction. Another class of AFs is derived to accommodate constraints by reducing a specific type of uncertainty measure about a quantity of interest based on the observations, which is known as stepwise uncertainty reduction [26]. As suggested in previous studies [26], many AFs can be derived to infer any quantity of interest, depending on different types of uncertainty measures. In [143], an uncertainty measure based on PI has been defined, where constraints are further accounted for by combining the probability of feasibility. Using the same principle, integrated expected conditional improvement in [15] defines the expected reduction in EI under the constrained satisfaction probabilities, allowing the unfeasible area to provide information. Another popular uncertainty measure is entropy inspired by information theory, which has been explored in [73, 142]. Hernández-Lobato *et al.* [71] extended PES to unknown constrained problems by introducing the conditional predictive distributions, with the assumption of the independent GP priors of the objective and constraints. A follow-up work [72] further investigated the use of PES in the presence of decoupled constraints, in which subsets of the objective and constraint functions can be evaluated independently. However, PES encounters the difficulty of calculation, which motivates the use of max-value entropy search for constrained problems in a recent work [142].

3.4.3 Multi-step look-ahead methods. Most AFs are myopic, called one-step look-ahead methods, as they greedily select locations for the next true evaluation, ignoring the impact of the current selection on the future steps. By contrast, few non-myopic AFs have been developed to select samples by maximizing the long-term reward from a multi-step look-ahead [206]. For example, Lam and Willcox [101] formulated the look-ahead BO as a dynamic programming (DP) problem, which is solved by an approximate DP approach called rollout. This work subsequently was extended to constrained BO by redefining the stage-reward as the reduction of the objective function satisfying the constraints [100]. The computation burden resulting from rollout triggers the most recent work by Zhang *et al.* [214], where a constrained two-step AF, called 2-OPT-C, has been proposed.

3.4.4 Surrogate-assisted constraint-handling methods. The above-mentioned constraint-handling techniques focus on the AFs within the BO framework, where a GP model generally serves as a global model. In the evolutionary computation community, many attempts have been made to combine the best of both worlds in the presence of expensive problems subject to constraints. One avenue is to use MOEAs to optimize the objectives and constraints simultaneously. For example, instead of maximizing the product of EI and the probability of feasibility, the two AFs can be served as two objectives and optimized by an MOEA, and a set of new samples are randomly selected from the obtained Pareto optimal candidates[211].

3.4.5 **Discussions.** Most constraint-handling BO methods are achieved by introducing new AFs, with a few attempts adopted augmented Lagrangian relaxation to convert the constrained optimization problems into simple unconstrained problems [60]. For highly constrained problems, it is difficult to construct surrogates with good quality for the entire search space due to limited or even unavailable feasible samples. A promising direction is to search the feasible region first, and then approach to the best feasible solution. For example, conducting both local and global search to accelerate the search for feasible points [85] is very promising, but further research is required. In many applications, evaluation costs, user's preference and fairness can be defined as constraints [141], which is an interesting future research direction.

3.5 Multi-objective optimization

Many real-world optimization problems have multiple conflicting objectives to be optimized simultaneously, which are referred to as multi-objective optimization problems (MOPs) [217]. Mathematically, an MOP can be formulated as

$$\begin{array}{ll} \min_{\mathbf{x}} & \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})) \\ \text{s.t.} & \mathbf{x} \in \mathcal{X} \end{array} \tag{18}$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_d)$ is the decision vector with d decision variables, $\mathcal{X}$ denotes the decision space, and the objective vector $\mathbf{f}$ consists of m ($m \geq 2$) objectives. Note that for many-objective problems (MaOPs) [107], the number of objectives m is larger than three. Here the target is to find a set of optimal solutions that trade off between different objectives, which are known as Pareto optimal solutions. The whole set of Pareto optimal solutions in the decision space is called Pareto set (PS), and the projection of PS in the objective space is called Pareto front (PF). The aim of multi-objective optimization is to find a representative subset of the Pareto front and MOEAs have been shown to be successful to tackle MOPs [217].

The objective functions in an MOP can be either time-consuming or costly. Thus, only a small number of fitness evaluations is affordable, making plain MOEAs hardly practical. Recall that GPs and AFs in BO are designed for single-objective black-box problems, therefore new challenges arise when BO is extended to MOPs, where sampling of multiple objective functions needs to be determined, and both accuracy and diversity of the obtained solution set must be taken into account. To meet these challenges, multi-objective BO is proposed by either embedding BO into MOEAs or converting an MOP into single-objective problems. Multi-objective BO can be largely divided into three categories: combinations of BO with MOEAs, performance indicator-based AFs, and information theory based AFs. Note that some of them may overlap and are thus not completely separable.

3.5.1 **Combinations of Bayesian optimization with MOEAs.** Since MOEAs have been successful in solving MOPs, it is straightforward to combine Bayesian optimization with MOEAs. This way, GPs and existing AFs for single-objective optimization can be directly applied to each objective in MOPs. According to the way in which Bayesian optimization and evolutionary algorithms work together, the combinations can be further divided into two groups, evolutionary Bayesian optimization (EBO) and Bayesian evolutionary optimization (BEO) [148]. In EBO, as shown in Fig. 2 (a) Bayesian optimization is the basic framework in which the AF is optimized using an evolutionary algorithm. By contrast, in BEO, as shown in Fig. 2 (b), the evolutionary algorithm is the basic framework, where the AF is adopted as a criterion for selecting offspring individuals to be sampled. However, the objective functions in environmental selection of the MOEA may be different from the AFs. The differences that distinguish these methods lie in the adopted MOEAs and the strategy for selecting new samples. Typically, decomposition based MOEAs use a scalarizing function, such as the Tchebycheff scalarizing function or the weighted sum, to generate a set of single-objective problems. ParEGO [98] is an early EBO in this category: the augmented Tchebycheff function with a set of randomly generated weight vectors is adopted to construct

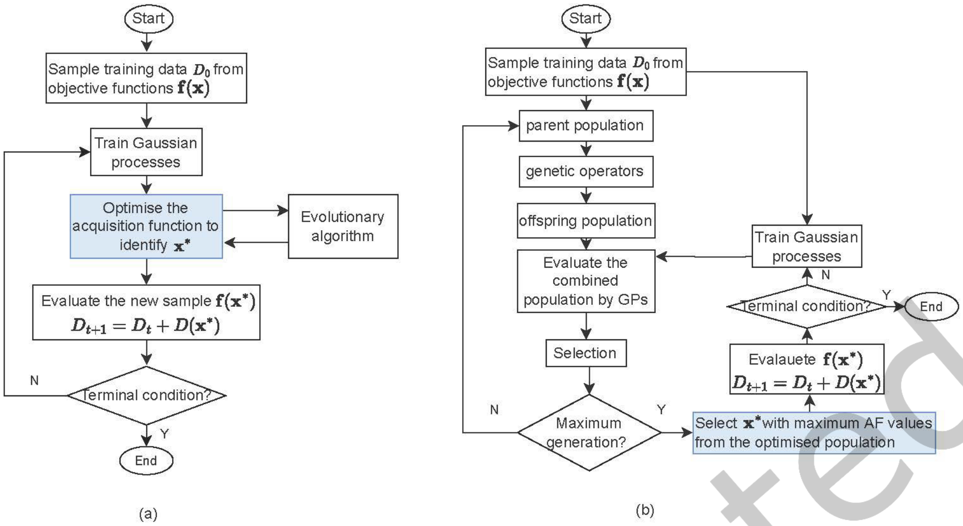
Start
Sample training data D_0 from objective functions $\mathbf{f}(\mathbf{x})$
Train Gaussian processes
Optimise the acquisition function to identify $\mathbf{x}^*$
Evolutionary algorithm
Evaluate the new sample $\mathbf{f}(\mathbf{x}^*)$
$D_{t+1} = D_t + D(\mathbf{x}^*)$
N
Terminal condition?
Y
End
(a)
Start
Sample training data D_0 from objective functions $\mathbf{f}(\mathbf{x})$
parent population
genetic operators
offspring population
Train Gaussian processes
Evaluate the combined population by GPs
N
Terminal condition?
Y
End
Selection
Evalauete $\mathbf{f}(\mathbf{x}^*)$
$D_{t+1} = D_t + D(\mathbf{x}^*)$
N
Maximum generation?
Y
Select $\mathbf{x}^*$with maximum AF values from the optimised population
(b)

Fig. 2. Two main approaches combining between evolutionary algorithms with Bayesian optimization: (a) evolutionary Bayesian optimization, and (b) Bayesian evolutionary optimization. In (b), the fitness functions for environmental selection in the evolutionary algorithm may be different from the acquisition function in infilling samples.

multiple single-objective optimization problems, to which the traditional AFs can be directly applied to identify new samples. By contrast, an MOP can be decomposed into multiple single-objective sub-problems, as done in the multiobjective evolutionary algorithm based on decomposition (MOEA/D) [209] and the reference vector guided evolutionary algorithm (RVEA) [25]. After that, Bayesian optimization can be applied to solve the sub-problems. For example, an EBO method, MOEA/D-EGO [210], uses the Tchebycheff scalarizing function to decompose an MOP into a set of single-objective subproblems and selects a set of new samples from the population by optimizing EI. Alternatively, a BEO method, Kriging-assisted RVEA (K-RVEA) [27], decomposes the MOP into a number of sub-problems using reference vectors. Then, the most uncertain solution is selected for sampling for each sub-problem if the diversity of the overall population needs to be promoted; otherwise, the solution having the best penalized angle distance according to the predicted objective values will be selected for each sub-problem. RVEA is also adopted as the optimizer in [186] to address expensive MOPs, where the predicted objective value and the uncertainty are weighted together as an AF, and the weights are tuned to balance exploration and exploitation.

Non-dominated sorting is another approach widely adopted in MOEAs. For example, Shinkyu *et al* [84] proposed an extension of EGO using a non-dominated sorting based MOEA (Multi-EGO), which is an EBO method. Multi-EGO maximizes the EIs for all objectives simultaneously, thus the non-dominated sorting is employed to select new samples. In recent work [12, 156], non-dominated sorting is used to select a cheap Pareto front based on the surrogate models. Similarly, multi-objective particle swarm optimization using non-dominated sorting is adopted in [109, 115] in combination with Bayesian optimization.

3.5.2 Performance indicator-based AFs. Performance indicators were originally developed to assess and compare the quality of solution sets (rather than a single solution) obtained by different algorithms [221]. Various quality indicators have been proposed, including inverted generational distance [216] and hypervolume (HV) [220]. HV calculates the volume of the objective space dominated by a set of non-dominated solutions $\mathcal{P}$ and bounded by a reference point $\mathbf{r}$,

$$\mathrm{HV}(\mathcal{P}) = VOL\left(\cup_{\mathbf{y}\in\mathcal{P}}[\mathbf{y},\mathbf{r}]\right) \tag{19}$$

where $VOL(\cdot)$ denotes the usual Lebesgue measure, $[\mathbf{y},\mathbf{r}]$ represents the hyper-rectangle bounded by y and r. Hence, algorithms achieving a larger HV value are better.

Interestingly, performance indicators can be incorporated into MOEAs in different manners. They can be adopted as an optimization criterion in the environmental selection [199] since they provide an alternative way to reduce an MOP into a single-objective problem. For this reason, various multi-objective Bayesian optimization methods with a performance indicator-based AF have been developed, among which HV is the most commonly used performance indicator. An early work is $\mathcal{S}$-Metric-Selection-based efficient global optimization (SMS-EGO) [146], which is based on the $\mathcal{S}$ metric or HV metric. In SMS-EGO, a Kriging model is built for each objective, then HV is optimized to select new samples, where the LCB is adopted to calculate the fitness values. Similarly, TSEMO [20] uses Thompson sampling on the GP posterior as an AF, optimizes multiple objectives with NSGA-II, and then selects the next batch of samples by maximizing HV.

Indeed, the combination of the EI and HV, which is known as expected hypervolume improvement (EHVI), is more commonly seen in the context of expensive MOPs. Given the current PF approximation $\mathcal{P}$, the contribution of a non-dominated solution $(\mathbf{x},\mathbf{y})$ to HV can be calculated by

$$I(\mathbf{y},\mathcal{P}) = HV(\mathcal{P}\cup\{\mathbf{y}\}) - HV(\mathcal{P}), \tag{20}$$

The EHVI quantifies the expectation of the HV over the non-dominated area. Hence, the generalized formulation of EHVI is formulated as

$$\mathrm{EHVI}(\mathbf{x}) = \int_{\mathbb{R}^m} I(\mathbf{y},\mathcal{P}) \prod_{i=1}^{m} \frac{1}{\sigma_i(\mathbf{x})}\phi\left(\frac{y_i(\mathbf{x})-\mu_i(\mathbf{x})}{\sigma_i(\mathbf{x})}\right) \mathrm{d}y_i(\mathbf{x}). \tag{21}$$

EHVI was first introduced in [42] to provide a scalar measure of improvement for prescreening solutions, and then became popular for handling expensive MOPs [110, 202]. Wagner *et al.* [184] studied different AFs for MOPs, indicating that EHVI has desirable theoretical properties. The comparison between the EHVI with other criteria [165], such as EI and estimation of objective values shows that EHVI maintains a good balance between the accuracy of surrogates and the exploration of the optimization. Despite the promising performance, the calculation of EHVI itself is computationally intensive due to the integral involved, limiting its application to MOPs/MaOPs. A variety of studies have been done to enhance the computation efficiency for EHVI. In [42], Monte Carlo integration is adopted to approximate the EHVI. Emmerich *et al.* [41] introduced a direct computation procedure for EHVI, which partitions the integration region into a set of interval boxes. However, the number of interval boxes scales at least exponentially with the number of Pareto solutions and objectives. In a follow-up work, Couckuyt *et al.* [29] introduced an efficient way by reducing the number of the interval boxes. Another commonly used indicator is based on distance, especially the Euclidean distance. Expected Euclidean distance improvement (EEuI) [95] defines the product of the probability improvement function and an Euclidean distance-based improvement function for a closed-form expression of a bi-objective optimization problem. A fast calculation method for EEuI is proposed using the Walking Fish Group algorithm [29]. Alternatively, the maximin distance improvement is adopted as the improvement function in [175].

3.5.3 Information theory based AFs.

Given the popularity of information theoretic approaches in the context of single-objective Bayesian optimization, it is not surprising that many information-based AFs for tackling expensive MOPs have been proposed. For example, PES is adopted to address MOPs, called PESMO [69]. However, optimizing PESMO is a non-trivial task: a set of approximations are performed; thus the accuracy and efficiency of PESMO can degrade. A subsequent work is the extension of the output-space-entropy based AF in the context of MOPs, known as MESMO [11]. Empirical results show that MESMO is more efficient than the PESMO. As pointed out in [174], MESMO fails to capture the trade-off relations among objectives for MOPs where no points in the PF are near the maximum of each objective. To fix this problem, Suzuki *at al.* [174] proposed a Pareto-frontier entropy search that considers the entire PF, in which the information gain is formulated as

$$I(\mathcal{F}^*; \mathrm{y} \mid \mathcal{D}_n) \approx H\ \ p(\mathrm{y} \mid \mathcal{D}_n)\ \ - \mathbb{E}_{\mathcal{F}^*}\ \ H\ \ p(\mathrm{y} \mid \mathcal{D}_n, \mathrm{y} \preceq \mathcal{F}^*) \tag{22}$$

where $\mathcal{F}^*$ is the Pareto front, $\mathrm{y} \preceq \mathcal{F}^*$ denotes y is dominated or equal to at least one point in $\mathcal{F}^*$.

3.5.4 Discussions.

BO methods for expensive MOPs mainly focus on the design of AFs and their applications are generally limited to low-dimensional MOPs due to the scalability issue of GPs and the high computational complexity of some AFs. Hence, possible future directions include the investigation of alternatives of GPs and the effective AFs for high-dimensional MOPs/MaOPs. Moreover, due to the PS/PF in MOPs, more efforts should be devoted to the selection of new samples in terms of balancing exploration and exploitation.

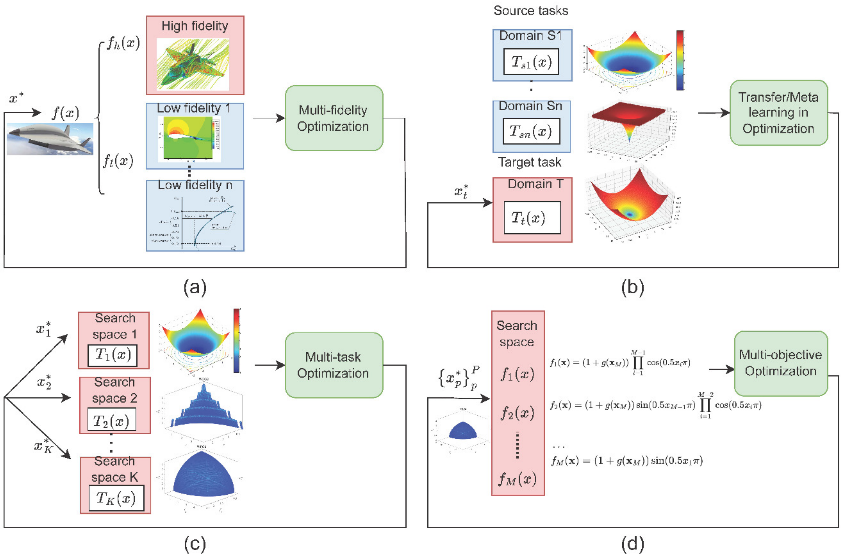

Fig. 3. The main difference between (a) multi-fidelity optimization, (b) transfer/meta learning in optimization, (c) multi-task optimization, and (d) multi-objective optimization. The target optimization task (denoted by red rectangles) in the four scenarios are different: while multi-objective optimization and multi-task optimization aim to effectively and concurrently optimize several problems, multi-fidelity optimization and transfer/meta learning aim to accelerate the target optimization task by utilizing useful knowledge acquired from low fidelity simulations or similar source optimization tasks (denoted by blue rectangles). In multi-task optimization, all tasks are equally important and knowledge transfer occurs between any of the related tasks. Finally, the difference between multi-objective optimization and multi-task optimization is that the former handles conflicting objectives of the same task, while each task in the latter can be a single/multi-objective problem.

3.6 Multi-task Optimization

Many black-box optimization problems are not one-off tasks. Instead, several related instances of the tasks can be simultaneously tackled, which is known as multi-task optimization. Suppose there are K optimization tasks, $i = \{1, 2, \ldots, K\}$, to be accomplished. Specifically, denote T_i as the i-th task to be optimized and X_i as the search space of T_i. Without loss of generality, assuming each task is a minimization problem, and multi-task optimization (MTO) aims to find a set of solutions $(\mathbf{x}_1^*, \ldots, \mathbf{x}_K^*)$ satisfying

$$\mathbf{x}_i^* = \underset{\mathbf{x} \in X_\mathbf{i}}{\arg\min} T_i(\mathbf{x}), i = 1, 2, \ldots, K. \tag{23}$$

There exist some conceptual similarities and overlaps between multi-task optimization and some other terms, such as multi-objective optimization, multi-fidelity optimization and transfer/meta learning. Similarities and differences are illustrated in Fig. 3. While multi-fidelity optimization and transfer/meta learning focus on the target task (referred to as asymmetric dependency structure), MTO treats all tasks equal and knowledge transfer occurs between any related tasks (referred to as symmetric dependency structure) [111]. Multi-task Bayesian optimization aims to optimize a collection of related tasks at the same time, thereby speeding up the optimization process by taking advantage of the common information across the tasks. There are two requirements to achieve this. First, surrogate models that can learn the transferable knowledge between the tasks should be built. Second, the AF should consider not only the exploration-exploitation balance, but also the correlation between the tasks, so that the data efficiency of optimization can be further improved by transferring knowledge between the related tasks. In the following, we present Bayesian optimization algorithms in which multi-task Gaussian models are constructed and specific AFs are designed for MTO.

3.6.1 **Multi-task Gaussian process.** MTO benefits from transferring knowledge across different tasks assuming that the tasks are related to a certain degree. In the geostatistics community, the linear model of coregionalization (LMC) expresses the outputs as linear combinations of Q independent random functions,

$$T_i(\mathbf{x}) = \sum_{q=1}^{Q} a_{i,q} u_q(\mathbf{x}), \tag{24}$$

where the latent function $u_q(\mathbf{x})$ is assumed to be a zero-mean Gaussian process with covariance as $k_q(\mathbf{X}, \mathbf{X}')$, and $a_{i,q}$ is the coefficient for $u_q(\mathbf{x})$. In the context of machine learning, many Bayesian multi-task models can be viewed as variations of the LMC with different parameterizations and constraints. A representative work is called multi-task GP [195], which uses the intrinsic coregionalization model kernel. Besides the covariance function over inputs $k_\mathcal{X}(\mathbf{x}, \mathbf{x}')$, a task covariance matrix $k_\mathcal{T}(t, t')$ is introduced as coregionalization metrics to model the inter-task similarities. Consequently, the product kernel can be derived as follows:

$$k((\mathbf{x}, t), (\mathbf{x}', t')) = k_\mathcal{X}(\mathbf{x}, \mathbf{x}') \otimes k_\mathcal{T}(t, t') \tag{25}$$

where $\otimes$ denotes the Kronecker product, and $t, t' \in \mathcal{T}$, $k_\mathcal{T}(t, t')$ is a positive semi-definite matrix, which is guaranteed by the Cholesky decomposition. The multi-task GP suffers from a high computational complexity of $O(K^3n^3)$. To address the scalability of multi-task GP, Matheron's rule is used to exploit the Kronecker structure in the covariance matrices to achieve faster predictive computations in [116]. In LMC models, the correlated process is expressed by a linear combination of a set of independent processes. Such a method is limited to scenarios where one output process is a blurred version of the other. Alternatively, convolution processes are employed to account for correlations across outputs, and each output can be expressed through a convolution integral between a smoothing kernel and a latent function [5].

3.6.2 **Acquisition functions in MTO.** Although many attempts have been made to propose multi-task models, only recently a few multi-task Bayesian optimization algorithms have been proposed, especially in the field of hyperparameter optimization in machine learning. Swersky and Snoek [177] extend the multi-task GP [195] to Bayesian optimization for knowledge transfer in tuning hyperparameters, where a new AF based on entropy search is proposed by taking cost into consideration. Similar ideas that adopt multi-task GPs or design a new AF introducing a trade-off between information gain and cost minimization can be found in [126]. Bardenet *et al.* [10] considered the hyper-parameter optimization for deep belief networks with different features of the dataset, and proposed collaborative tuning of several problems. In contextual policy search (CPS), a joint GP model over the context-parameter space is learned, allowing knowledge acquired from one context to be generalized to similar contexts. More recently, Thompson sampling has been extended to multi-task optimization by sampling from the posterior to identify the next task and action [22], which is theoretically guaranteed.

3.6.3 Discussions. Regarding the surrogate modeling of MTO, the commonly used LMC model is criticized for its computational complexity. While some simple models are proposed to alleviate this issue, their prediction qualities can be affected. Hence, the development of effective surrogate models for MTO is a promising direction. Indeed, a few attempts have been made to address MTO by defining new AFs, most of which consider one target task. In the future, it would be beneficial to select new samples for simultaneously optimizing all tasks.

3.7 Multi-fidelity optimization

Bayesian optimization generally assumes that only the target expensive objective function is available, which is referred to as single-fidelity optimization. In many practical problems, however, the evaluation of the target function $f(\mathbf{x})$ can often be run at multiple levels of fidelity with varying costs, $f_1(\mathbf{x}), \dots, f_M(\mathbf{x})$, where the higher the fidelity $m \in \{1, 2, \dots, M\}$, the more accurate but costly the evaluation will be. This is known as multi-fidelity optimization (MFO), which can be seen as a subclass of multi-task learning, where the group of related functions can be meaningfully ordered by their similarity to the objective function.

MFO aims to accelerate the optimization of the target objective and reduce the optimization cost by jointly learning the maximum amount of information from all fidelity models. To achieve this, Bayesian optimization undertakes two changes to make use of multiple fidelity data, namely multi-fidelity modeling and a new sample selection, which will be discussed in detailed in the following.

3.7.1 Multi-fidelity models. Typically, multi-fidelity Bayesian optimization builds surrogate models of different levels of fidelity either by learning an independent GP for each fidelity [89], or jointly modeling multi-fidelity data to capture the correlation between the different fidelity data, such as multi-output GP and deep neural networks. Among them, one most popular multi-fidelity model is Co-Kriging [128]. Kennedy and O'Hagan [96] proposed an autoregressive model to approximate the expensive high-fidelity simulation $\hat{y}_H(\mathbf{x})$ by the sum of the low-fidelity Kriging model $\hat{y}_L(\mathbf{x})$ and a discrepancy model $\hat{\delta}(\mathbf{x})$, formulated as

$$\hat{y}_H(\mathbf{x}) = \rho \hat{y}_L(\mathbf{x}) + \hat{\delta}(\mathbf{x}) \tag{26}$$

where ρ denotes a scaling factor minimizing the discrepancy between $\rho \hat{y}_L(\mathbf{x})$ and high-fidelity model at the common sampling points. Thus, high-fidelity model can be enhanced by acquiring information from the low-fidelity inexpensive data. Later, a Bayesian hierarchical GP model is developed in [147] to account for complex scale changes from low fidelity to high fidelity. To improve the computational efficiency, a recursive formulation for Co-Kriging was proposed in [103], assuming that the training datasets for $\hat{y}_H(\mathbf{x})$ and $\hat{y}_L(\mathbf{x})$ have a nested structure, i.e., the training data for the higher fidelity levels is a subset of that of a lower fidelity level. Hence, the GP prior $\hat{y}_L(\mathbf{x})$ in Eq. 26 is replaced by the corresponding GP posterior, improving the efficiency of the hyperparameter estimations. Following this idea, the autoregressive multi-fidelity model given in Eq. 26 has been generalized by replacing the scaling factor ρ with a non-linear mapping function [140]. Alternatively, multi-fidelity deep GP models use a neural network to learn a non-linear transformation [31], which is further extended to different input spaces in terms of parametrization forms and dimensionality [67].

3.7.2 Acquisition functions for multi-fidelity optimization. Based on multi-task models [96, 103], the design of sophisticated AFs to select both the input locations and the fidelity in the MFO setting has attracted much research interest. Earlier multi-fidelity AFs focused on the adaptation of EI. Huang *et al.* [78] proposed an augmented EI function to account for different fidelity levels of an infill point. Specifically, the proposed EI is the product of the expectation term, the correlation between the low-fidelity and high-fidelity models, the ratio of the reduction in the posterior standard deviation after a new replicate is added [79], and the ratio between the evaluation cost of the different fidelity models. To enhance the exploration capability of augmented EI, Liu *et al.* [114] proposed a sample density function that quantifies the distance between the inputs to avoid clustered samples. UCB has been widely used in MFO, especially in bandit problems. An early work on principled AF based UCB for MFO is MF-GP-UCB [89]. The MF-GP-UCB algorithm first formulates an upper bound for each fidelity, among which the minimum bound is identified to be maximized for selecting the new sample. Having selected the new point, a threshold is introduced to decide which fidelity to query. In a follow-up work [91], MF-GP-UCB is extended to the continuous fidelity space. Sen *et al.* [162] developed an algorithm based on a hierarchical tree-like partitioning, and employed MF-GP-UCB to select the leaves. The motivation behind this method is to explore coarser partitions at lower fidelities and proceed to finer partitions at higher fidelities when the uncertainty has shrunk. Following this idea, Kandasamy *et al.* [90] adopted MF-GP-UCB to explore the search space at lower fidelities, and then exploit the high fidelities in successively smaller regions. Recently, information-theoretic approaches have become popular in MFO. For example, ES with the Co-Kriging model is adopted in [117] to solve a two-fidelity optimization. In [213], unknown functions with varying fidelities are jointly modeled as a convolved Gaussian process [5], then a multi-output random feature approximation is introduced to calculate PES. Since it is non-trivial to calculate the multi-fidelity AFs based on ES/PES, MES has been extended to MFO due to its high computational efficiency [178].

3.7.3 Discussions. The multi-fidelity models generally require strong assumptions: the low fidelity and high fidelity are always linearly correlated and the search spaces are the same. These assumptions may not hold in real-world applications, such as varying search space dimensions for different fidelities. More efforts should be devoted to the exploration of alternative models. For the AFs in MFO, there lack investigations for the continuous-fidelity setting. Moreover, existing multi-fidelity BO techniques mainly address bandit problems and single-objective problems, it is therefore interesting to extend them to MOPs and robust optimization.

3.8 Transfer/Meta Learning

Although Bayesian optimization offers a powerful data-efficient approach to global black-box optimization problems, it considers each task separately and often starts a search from scratch, which needs a sufficient number of expensive evaluations before achieving high-performance solutions. To combat such a "cold start" issue, transfer/meta learning in Bayesian optimization has attracted a surge of interest in recent years. Given a set of auxiliary/source domains D_s and optimization tasks T_s, a target domain D_T and optimization task T_T, transfer/meta learning in Bayesian optimization aims to leverage knowledge from previous related tasks T_s to speed up the optimization for the target task T_T. A well-studied example is hyperparameter optimization of a machine learning algorithm on a new dataset (target) with observed hyperparameter performances on the other datasets (source/meta-data). The availability of meta-data from previous related tasks in hyperparameter optimization has motivated meta-initialization to initialize a hyperparameter search based on the best hyperparameter configurations for similar datasets [45]. Typically, the two terms, i.e., transfer/meta learning, are used interchangeably in the context of Bayesian optimization. Note that in the BO community, knowledge transfer has also been investigated under the several umbrellas, including multi-task learning and multi-fidelity optimization, which may overlap with the broad field of transfer learning. According to the method for capturing the similarity, we classify the Bayesian optimization algorithms coupled with transfer learning techniques into the following three groups.

3.8.1 Hierarchical model. Hierarchical models learned across the entire datasets arise as a natural solution to making use of the knowledge from related source domains. For example, Bardenet *et al.* [10] noted that the loss values on different datasets may differ in scale, motivating a ranking surrogate to map observations from all runs into the same scale. However, this approach suffers from a high computational complexity incurred by the ranking algorithm. To address this problem, Yogatama and Mann [204] suggested to reconstruct the response values by subtracting the per-dataset mean and scaling through the standard deviation, while Golovin *et al.* [56] proposed an efficient hierarchical GP model using the source posterior mean as the prior mean for the target.

3.8.2 Multi-task Gaussian process.

Since multi-task GP models are powerful for capturing the similarity between the source and target tasks, Swersky *et al.* [177] conducted a straightforward knowledge transfer using a multi-task GP. Meanwhile, the positive semi-definite matrix in multi-task GPs (see Eq. 25) has been modified to improve the computational efficiency [122, 204]. On the other hand, Joy *et al.* [87] assumed that the source data are noisy observations of the target task, so that the difference between the source and target can be modeled by noise variances. Following this idea, Ramachandran *et al.* [149] further improved the efficiency of the knowledge transfer by using a multi-bandit algorithm to identify the optimal source.

3.8.3 Weighted combination of GPs.

Knowledge transfer in Bayesian optimization can also be achieved by a weighted combination of GPs. Instead of training a single surrogate model on a large training data set (i.e., the historical data), Schilling *et al.* [159] suggested to use the product of GP experts to improve the learning performance. Specifically, an individual GP is learned on each distinct dataset. This way, the prediction on a target data provided by the product of the individual GPs is a sum of means with weights adjusted with regard to the GP uncertainty. Different strategies have been proposed to adapt the weights in the combination [44]. In multi-objective optimization, Min *et al.* [123] proposed to identify the weights by optimizing the squared error of out-of-sample predictions. In a complementary direction, a few attempts have been dedicated to leveraging the meta-data within the AF in a similar fashion to the weighted combination of GPs. A representative work is called transfer AF [197], which is defined by the weighted average of the expected improvement on the target dataset and source datasets. More recently, Volpp *et al.* [183] adopted reinforcement learning to achieve this.

3.8.4 Discussions.

Intuitively, the optimization of the target task may suffer from negative transfer if the learned knowledge degrades the performance. Hence, the surrogate model that captures the similarity between target and auxiliary tasks and how to alleviate the negative transfer remain active fields of research. Generally, there is an implicit assumption that the source and target domains share the same search spaces, which greatly limits their applications. In the future, the heterogeneous search spaces should be investigated. Moreover, it is interesting to protect the data privacy during knowledge transfer.

3.9 Parallel/Batch Bayesian optimization

The canonical Bayesian optimization is inherently a sequential process since one new data is sampled in each iteration, which might be inefficient in many applications where multiple data points can be sampled in parallel [132]. A strength of sequential Bayesian optimization is that a new data point is selected using the maximum available information owing to the immediately updated GP, and therefore searching for multiple query points simultaneously is more challenging. With the growing availability of parallel computing, an increasing number of studies exploring batch Bayesian optimization have been carried out, which can be roughly classified into two groups. One is the extension of the existing AFs to batch selection, and the other is problem reformulation.

3.9.1 Extensions of the existing AFs.

A pioneering multi-points AF is the parallelized version of the EI, called q-points EI (q-EI) [53, 54]. The q-EI is straightforwardly defined as the expected improvement of the q points beyond the current best observation. However, the exact calculation of q-EI depends on the integral of q-dimensional Gaussian density, and therefore becomes intractable and intensive as q increases. Hence, Ginsbourger *et al.* [53] sequentially identified q points by using Kriging believer or constant liar strategies to replace the unknown output at the last selected point, facilitating the batch selection based on q-EI. Treatments for the intractable calculation of q-EI have been investigated in [54, 185]. Besides, an asynchronous version of q-EI is presented in [82].

The parallel extension of the GP-UCB has been widely investigated owing to its theoretical guarantees, i.e., the sublinear growth of cumulative regret. An extension of GP-UCB is proposed to leverage the updated variance, encouraging more exploration [35]. Similarly, a GP-UCB approach with pure exploration is proposed in [28], which identifies the first query point via the GP-UCB, while the remaining ones are selected by maximizing the updated variance. Since MOEAs can provide a set of non-dominated recommendations, they are well-suited for determining the remaining points by simultaneously optimizing the predicted mean and variance [64]. More diverse batches can be probed by sampling from determinantal point processes (DPPs) [193]. With the rapidly growing interest in batch Bayesian optimization, more AFs have been extended to the parallel setting. For example, parallelized PES [163] and KG [198] are developed to jointly identify a batch of points to probe in the next iteration, rendering, however, a poor scalability to the batch size. Interestingly, a state-of-the-art information-based AF, called trusted-maximizers entropy search, is proposed by introducing trusted maximizers to simplify the information measure [131], which is well scalable to the batch size. TS can also be extended to the parallel setting by sampling q functions instead [74]. More recently, TS has attracted much attention, as the inherent randomness of TS automatically achieves a balance between exploitation and exploration [92]. Note that the performance of TS is not necessarily better than traditional AFs, such as EI and UCB.

3.9.2 Problem reformulation.

Much effort has been devoted to developing new batch approaches by reformulating the optimization problem of AFs in parallel Bayesian optimization. One interesting direction aims to develop new batch AFs to select input batches that closely match the expected recommendation of sequential methods. For example, a batch objective function minimizing the loss between the sequential selection and the batch is defined in [7], which corresponds to a weighted k-means clustering problem. Given that the sequentially selected inputs are sufficiently different from each other, a maximization-penalization strategy is introduced by adding a local penalty to the AF [59]. Liu *et al.* [113] applied a multi-start strategy and gradient-based optimizer to optimize the AF, aiming to identify the local maxima of the AF. In addition, the multi-objective optimizer is a promising approach to finding a batch of query points [212], particularly for addressing expensive MOPs [27, 186]. Similarly, sequentially optimizing multiple AFs is amenable to generating batches of query points [88]. To better balance exploration and exploitation, different selection metrics can be combined [58, 77].

3.9.3 Discussions.

The major challenge for the design of new AFs in batch selection is the requirement of maximizing the information gain while avoiding the redundancy. Moreover, the scalability to the batch size is expected to be further investigated. As batch BO can be employed in many real-world applications, it is interesting to consider more practical problem settings, such as high-dimensional search spaces and asynchronously parallel settings.

4 CHALLENGES AND FUTURE DIRECTIONS

BO is a well-established powerful optimization method for handling expensive black-box problems, which has found many successful real-world applications. Despite all these advances, numerous challenges remain open. In fact, the field of Bayesian optimization keeps very active and dynamic, partly because an increasing number of new applications in science and technology poses new challenges and demands. In the following, we present several most recent important developments in Bayesian optimization and discuss future research directions according to the nature of optimization problems and settings, including but not limited to distributed, federated BO, dynamic optimization, heterogeneous evaluations, algorithmic fairness and non-stationary optimization.

4.1 Distributed Bayesian optimization

Despite a proliferation of studies on parallel or batch Bayesian optimization in recent years, most of them require a central server to construct a single surrogate model with few exceptions. Distributed Bayesian optimization has emerged to handle distributed optimization, where the search space, the sampling process, the expensive evaluations and GPs can be distributed. For example, a straightforward distributed Bayesian optimization, called HyperSpace, has been proposed by Young *et al.* [205] for hyperparameter optimization. HyperSpace partitions the large search space with a degree of overlap and all possible combinations of these hyperspaces are generated and equipped with a GP model, allowing us to run the optimization loop in parallel. Thompson sampling can be fully distributed and handle the asynchronously parallel setting [75], although it fails to perform well due to its inherent randomness. Barcos and Cantin [49] presented an interpretation of Bayesian optimization from the Markov decision process perspective and adopted Boltzmann/Gibbs policy to select the next query, which can be performed in a fully distributed manner.

Several questions remain open in design of distributed Bayesian optimization. First, it is of fundamental importance to achieve a trade-off between the convergence rate and communication cost. The convergence of distributed Bayesian optimization needs more rigorous theoretical proof and requires further improvement, and the computational gains will be offset in the presence of communication latencies. Second, it is still barely studied how to handle asynchronous settings that result from time-varying communication costs, different computation capabilities and heterogeneous evaluation times. Third, it is an important yet challenging future direction to take more practical scenarios into consideration, such as complex communication networks and communication constraints.

4.2 Federated Bayesian optimization

While the rapidly growing sensing, storage and computational capability of edge devices has made it possible to train powerful deep models, increasing concern over data privacy has motivated a privacy-preserving decentralized learning paradigm, called federated learning [121]. The basic idea in federated learning is that the raw data remains on each client, while models trained on the local data are uploaded to a server to be aggregated, thereby preserving the data privacy. Adapting Bayesian optimization to the federated learning setting is motivated by the presence of black-box expensive machine learning and optimization problems.

Dai *et al.* [32] explored the application of Bayesian optimization in the horizontal federated learning setting, where all agents share the same set of features and their objective functions are defined on a same domain. Federated TS (FTS), which samples from the current GP posterior on the server with a probability of p and consequently samples from the GP provided by the clients with a probability $1 - p$. However, FTS lacks a rigorous privacy guarantee. To remedy this drawback, differential privacy [40], a mathematically rigorous approach to privacy preservation, is introduced into FTS, called DP-FTS [33]. Instead of using GPs as surrogates, Xu *et al.* [201] proposed to use radial-basis-function networks (RBFNs) on local clients. A sorting averaging strategy is proposed to construct a global surrogate on the server, where each local RBFN is sorted by a matching metric, and the parameters of each local surrogate are averaged according to the sorted index. The RBFN-based federated optimization was extended to handle multi/many-objective optimization problems [200]. Although much work addressing challenges in federated learning, including communication efficiency, systems and data heterogeneity, and privacy protection have been reported, privacy-preserving optimization brings with many new questions. First, since GP is non-parameter models, it cannot be directly applied to the federated setting. One idea is to approximate the GP model with random Fourier feature approximates [32], in which representative power and computation efficiency should be taken into consideration. Second, Thompson sampling is adopted as AF due to its ability to handle heterogeneous settings; however, it is criticized by its poor performance compared with other AFs. Hence, further investigation in new acquisition methods is an interesting yet challenging research direction.

Finally, privacy protection in federated Bayesian optimization remains elusive, and more rigorous definitions of threat models in the context of distributed optimization are highly demanded.

4.3 Dynamic optimization

In many real-world applications, such as network resource allocation, recommendation systems, and object tracking, the objective function to be optimized may change over time. Such optimization scenarios are known as dynamic optimization or time-dependent problems. Solving such problems is challenging for most optimization techniques designed for stationary problems [203]. Although various Bayesian optimization algorithms for solving static expensive black-box problems have been proposed, only a few methods have been developed to handle dynamic optimization problems.

Most Bayesian optimization methods for dynamic optimization rely on the multi-armed bandit (MAB) setting with time-varying reward functions. MAB models the sequential decision-making with partial information, where the gambler requires to choose one of the K slot machine arms at each iteration in order to maximize the cumulative reward [215]. Bogunovic *et al.* [19] introduced a simple Markov model for the reward functions using GPs, allowing the GP model to vary at a steady rate. Instead of treating all the samples equally important, *resetting* [215], *temporal kernel* [24], *sliding window* [218], and *weighted GP model* [34] have been proposed to achieve forgetting-remembering trade-off. Nevertheless, the construction of effective surrogates for time-dependent objective functions, the design of AFs to identify promising solutions and track the optimum remain challenging problems. Moreover, it is interesting to incorporate advances in machine learning, such as transfer learning, for leveraging informative from the previous runs.

4.4 Heterogeneous evaluations

Bayesian optimization implicitly assumes that the evaluation cost in different regions of the search space is the same. This assumption, however, can be violated in practice. For example, the evaluation times of different hyperarameter settings and the financial cost for steel or drug design using different ingredients [2] may vary dramatically. Moreover, in multi-objective optimization, different objectives may have significantly different computational complexities, known as heterogeneous objective functions [4]. Handling heterogeneous evaluation costs that arise in both search spaces and objective spaces has attracted increased attention, motivating the development of cost-aware Bayesian optimization.

Most cost-aware Bayesian optimization algorithms focus on single-objective optimization problems. Snoek *et al.* [168] introduces an AF called *expected improvement per second* to balance between the cost efficiency and evaluation quality via dividing EI by cost. This approach, however, tends to exhibit good performance only when the optimal solution is computationally cheap. In [104], an optimization problem constrained by a cost budget is formulated as a constrained Markov decision process and then a rollout AF with a number of look-ahead steps is proposed. To handle heterogeneous computational costs of different objectives in MOPs, simple *Interleaving schemes* are developed to fully utilize the available per-objective evaluation budget [4]. More recently, the search experience of cheap objectives is leveraged to help and accelerate the optimization of expensive ones, thereby enhancing the overall efficiency in solving the problem. For example, Wang *et al.* [189] made use of domain adaptation techniques to align the solutions on/near the Pareto front in a latent space, which allows data augmentation for GPs of the expensive objectives. Alternatively, a co-surrogate model is introduced to capture the relationship between the cheap and expensive objectives in [188]. Most recently, a new AF that takes both the search bias and the balance between exploration and exploitation into consideration was proposed [187], thereby reducing the search bias caused by different per-objective evaluation times in MOPs and MaOPs.

Bayesian optimization for heterogeneous settings is still a new research field. This is particularly true when there are many expensive objectives but their computational complexities significantly differ.

4.5 Algorithmic fairness

With the increasingly wider use of machine learning techniques in almost every field of science, technology and human life, there is a growing concern with the fairness of these algorithms. A large body of literature has demonstrated the necessity of avoiding discrimination and bias issues in finance, health care, hiring, and criminal justice that may result from the application of learning and optimization algorithms. A number of unfairness mitigation techniques have been dedicated to measuring and reducing bias/unfairness in different domains, which can be roughly divided into three groups, pre-, in-, and post processing, according to when the technique is applied [141]. The first group aims to re-balance the data distribution before training the model. The second group typically trains the model either under fairness constraints or combining accuracy metrics with fairness, while the third group adjust the model after the training process.

Accounting for fairness in the Bayesian optimization framework is a largely unexplored territory with few exceptions. For example, Perrone *et al.* [141] proposed an in-processing unfairness mitigation method in hyper-parameter optimization based on a constrained Bayesian optimization framework, called FairBO. In FairBO, an additional GP model is trained for the fairness constraint, allowing cEI to select new queries that satisfies the constraint. Unfortunately, such a constrained optimization method is designed for a single definition of fairness, which is not always applicable. A different fairness concept was developed in a collaborative Bayesian optimization setting [167], in which parties jointly optimize a black-box objective function. It is undesired for each collaborating party to receive unfair rewards while sharing their information with each other. Consequently, a new notion, called fair regret, is introduced based on fairness concepts from economics. Following the notion, the distributed batch GP-UCB is extended using a Gini social-evaluation function to balance the optimization efficiency and fairness.

The fairness problem in the context of Bayesian optimization is vital yet under-studied, and the measurement and mathematical definitions have not been explicit. Hence, the fairness definition should be well-defined at first, so that the fairness requirement can be more precisely integrated into the Bayesian optimization. The second fundamental open question is to investigate how fair surrogate models in Bayesian optimization are and how fair the selected new samples are. Finally, bias reduction strategies in Bayesian optimization can only be applied to the simplest case where a single fairness definition is adopted. The design of practical fairness-aware Bayesian optimization methods is still an open question.

4.6 Non-stationary Optimization

The standard Gaussian process generally adopts a stationary kernel function under the assumption that the covariance between two data points is invariant to translation. However, this assumption is susceptible to non-stationary functions that have different variability across its range, which is commonly encountered in a broad field, such as aerospace engineering, signal processing, and geostatistics [66, 137].

Many efforts have been dedicated to addressing this issue. First, non-stationary kernel functions based on kernel convolution have been proposed to achieve the input-independent lengthscale [137], resulting in the high parametrization requirements. Alternatively, local stationary approaches have been proposed to accommodate the non-stationary function by dividing the input space and fitting stationary models in each region [151]. However, this class of methods heavily rely on its separability. In addition, input space warpings or non-linear mappings are used to remove the non-stationary effects in a latent space [170]. More recently, Hebbal *et al.* [66] leveraged the flexibility of deep GPs resulting from the deep learning theory to approximate the non-stationary functions, but deep GPs are not analytically tractable and suffer from the approximated posterior distribution.

Although the need for non-stationary modeling is largely acknowledged in BO, it is still an open question to explore the non-stationary surrogate models. It is interesting to include more practical requirements, such as scaling the non-stationary models to high-dimensional problems. Deep GPs have shown promising performance on complex and non-stationary optimization [66]; however, it is necessary to provide a theoretical analysis in terms of the inference of the posterior distribution.

4.7 Negative Transfer

Multi-fidelity optimization, multi-task optimization and transfer/meta learning in BO aim to transfer useful information from related tasks to improve the BO search. However, transferring less related knowledge can hurt the BO performance, which is also known as negative transfer. Hence, the success of transfer learning is heavily conditioned on reducing the probability of negative transfer.

While the aforementioned algorithms have shed light on the effectiveness of the transfer optimization paradigm, circumventing negative transfer in remains an open question. There lacks a rigorous definition for negative transfer in BO, such as how to distinguish the negative and positive transfer. Moreover, systematic treatments and analysis deserve further investigations, including the criteria for measuring the similarity between domains or tasks and adaptive transfer learning.

5 CONCLUSION

Bayesian optimization has become a popular and efficient approach to solving black-box optimization problems, and new methods have been emerging over the last few decades. In this paper, we performed a systematic literature review on Bayesian optimization, focused on new techniques for building the GP model and designing new AFs to apply Bayesian optimization to various optimization scenarios. We divide these scenarios into nine categories according to the challenges in optimization, including high-dimensional decision and objective spaces, discontinuous search spaces, noise, constraints, and high computational complexity, as well as techniques for improving the efficiency of Bayesian optimization such as multi-task optimization, multi-fidelity optimization, knowledge transfer, and parallelization. Lastly, we summarize most recent developments in Bayesian optimization that address distributed data, data privacy, fairness in optimization, dynamism, and heterogeneity in the objective functions. So far, only sporadic research has been reported in these areas and many open questions remain to be explored.

We hope that this survey paper can help the readers get a clear understanding of research landscape of Bayesian optimization, including its motivation, strengths and limitations, and as well as the future directions that are worth further research efforts.

REFERENCES

[1] 2018. Approaches to robust process identification: A review and tutorial of probabilistic methods. *Journal of Process Control* 66 (2018), 68–83.

[2] Majid Abdolshah, Alistair Shilton, Santu Rana, Sunil Gupta, and Svetha Venkatesh. 2019. Cost-aware multi-objective Bayesian optimisation. In *Proceedings of ICML Workshop on Automated Machine Learning*.

[3] Shipra Agrawal and Navin Goyal. 2012. Analysis of Thompson Sampling for the Multi-armed Bandit Problem. In *Proceedings of the 25th Annual Conference on Learning Theory (Proceedings of Machine Learning Research, Vol. 23)*, Shie Mannor, Nathan Srebro, and Robert C. Williamson (Eds.). PMLR, Edinburgh, Scotland, 39.1–39.26.

[4] Richard Allmendinger, Julia Handl, and Joshua Knowles. 2015. Multiobjective optimization: When objectives exhibit non-uniform latencies. *European Journal of Operational Research* 243, 2 (2015), 497–513.

[5] Mauricio A Alvarez and Neil D Lawrence. 2011. Computationally efficient convolved multiple output Gaussian processes. *The Journal of Machine Learning Research* 12 (2011), 1459–1500.

[6] Rika Antonova, Akshara Rai, Tianyu Li, and Danica Kragic. 2020. Bayesian optimization in variational latent spaces with dynamic compression. In *Conference on Robot Learning*. PMLR, 456–465.

[7] Javad Azimi, Alan Fern, and Xiaoli Z Fern. 2010. Batch Bayesian optimization via simulation matching. In *Advances in Neural Information Processing Systems*. Citeseer, 109–117.

[8] Samineh Bagheri, Wolfgang Konen, Richard Allmendinger, Jürgen Branke, Kalyanmoy Deb, Jonathan Fieldsend, Domenico Quagliarella, and Karthik Sindhya. 2017. Constraint handling in efficient global optimization. In *Proceedings of the Genetic and Evolutionary*

Computation Conference. 673–680.

[9] Ricardo Baptista and Matthias Poloczek. 2018. Bayesian optimization of combinatorial structures. In *International Conference on Machine Learning*. PMLR, 462–471.

[10] Rémi Bardenet, Mátyás Brendel, Balázs Kégl, and Michele Sebag. 2013. Collaborative hyperparameter tuning. In *International conference on Machine Learning*. PMLR, 199–207.

[11] Syrine Belakaria and Aryan Deshwal. 2019. Max-value entropy search for multi-objective Bayesian optimization. In *International Conference on Neural Information Processing Systems (NeurIPS)*.

[12] Syrine Belakaria, Aryan Deshwal, Nitthilan Kannappan Jayakodi, and Janardhan Rao Doppa. 2020. Uncertainty-aware search framework for multi-objective Bayesian optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 10044–10052.

[13] Justin J Beland and Prasanth B Nair. 2017. Bayesian optimization under uncertainty. In *NIPS BayesOpt 2017 workshop*.

[14] James Bergstra, Dan Yamins, David D Cox, et al. 2013. Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms. In *Proceedings of the 12th Python in Science Conference*, Vol. 13. Citeseer, 20.

[15] J Bernardo, MJ Bayarri, JO Berger, AP Dawid, D Heckerman, AFM Smith, and M West. 2011. Optimization under unknown constraints. *Bayesian Statistics* 9, 9 (2011), 229.

[16] Mickael Binois and Nathan Wycoff. 2022. A survey on high-dimensional Gaussian process modeling with application to Bayesian optimization. *ACM Transactions on Evolutionary Learning and Optimization* 2, 2 (2022), 1–26.

[17] Laurens Bliek, Sicco Verwer, and Mathijs de Weerdt. 2021. Black-box combinatorial optimization using models with integer-valued minima. *Annals of Mathematics and Artificial Intelligence* 89, 7 (2021), 639–653.

[18] Ilija Bogunovic, Andreas Krause, and Jonathan Scarlett. 2020. Corruption-tolerant Gaussian process bandit optimization. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1071–1081.

[19] Ilija Bogunovic, Jonathan Scarlett, and Volkan Cevher. 2016. Time-varying Gaussian process bandit optimization. In *Artificial Intelligence and Statistics*. PMLR, 314–323.

[20] Eric Bradford, Artur M Schweidtmann, and Alexei Lapkin. 2018. Efficient multiobjective optimization employing Gaussian processes, spectral sampling and a genetic algorithm. *Journal of Global Optimization* 71, 2 (2018), 407–438.

[21] Eric Brochu, Vlad M Cora, and Nando De Freitas. 2009. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *Technical Report TR-2009-23, University of British Columbia, Computer Science.* (2009).

[22] Ian Char, Youngseog Chung, Willie Neiswanger, Kirthevasan Kandasamy, Andrew O Nelson, Mark Boyer, Egemen Kolemen, and Jeff Schneider. 2019. Offline contextual Bayesian optimization. *Advances in Neural Information Processing Systems* 32 (2019), 4627–4638.

[23] Bo Chen, Rui Castro, and Andreas Krause. 2012. Joint Optimization and Variable Selection of High-dimensional Gaussian Processes. In *Proceedings of the 29th International Conference on Machine Learning*. International Machine Learning Society, 1423–1430.

[24] Renzhi Chen and Ke Li. 2021. Transfer Bayesian Optimization for Expensive Black-Box Optimization in Dynamic Environment. In *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 1374–1379.

[25] Ran Cheng, Yaochu Jin, Markus Olhofer, and Bernhard Sendhoff. 2016. A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation* 20, 5 (2016), 773–791.

[26] Clément Chevalier, Julien Bect, David Ginsbourger, Emmanuel Vazquez, Victor Picheny, and Yann Richet. 2014. Fast parallel kriging-based stepwise uncertainty reduction with application to the identification of an excursion set. *Technometrics* 56, 4 (2014), 455–465.

[27] Tinkle Chugh, Yaochu Jin, Kaisa Miettinen, Jussi Hakanen, and Karthik Sindhya. 2016. A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization. *IEEE Transactions on Evolutionary Computation* 22, 1 (2016), 129–142.

[28] Emile Contal, David Buffoni, Alexandre Robicquet, and Nicolas Vayatis. 2013. Parallel Gaussian process optimization with upper confidence bound and pure exploration. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 225–240.

[29] Ivo Couckuyt, Dirk Deschrijver, and Tom Dhaene. 2014. Fast calculation of multiobjective probability of improvement and expected improvement criteria for Pareto optimization. *Journal of Global Optimization* 60, 3 (2014), 575–594.

[30] Noel Cressie. 1990. The origins of kriging. *Mathematical Geology* 22, 3 (1990), 239–252.

[31] Kurt Cutajar, Mark Pullin, Andreas Damianou, Javier González, and Neil Lawrence. 2018. Deep Gaussian processes for multi-fidelity modeling. In *NeurIPS 2018*.

[32] Zhongxiang Dai, Bryan Kian Hsiang Low, and Patrick Jaillet. 2020. Federated Bayesian optimization via Thompson sampling. *Advances in Neural Information Processing Systems* 33 (2020), 9687–9699.

[33] Zhongxiang Dai, Bryan Kian Hsiang Low, and Patrick Jaillet. 2021. Differentially private federated Bayesian optimization with distributed exploration. *Advances in Neural Information Processing Systems* 34 (2021).

[34] Yuntian Deng, Xingyu Zhou, Baekjin Kim, Ambuj Tewari, Abhishek Gupta, and Ness Shroff. 2022. Weighted Gaussian process bandits for non-stationary environments. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 6909–6932.

[35] Thomas Desautels, Andreas Krause, and Joel W Burdick. 2014. Parallelizing exploration-exploitation tradeoffs in Gaussian process bandit optimization. *Journal of Machine Learning Research* 15 (2014), 3873–3923.

[36] Aryan Deshwal, Syrine Belakaria, and Janardhan Rao Doppa. 2021. Mercer features for efficient combinatorial Bayesian optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 7210–7218.

[37] Aryan Deshwal, Syrine Belakaria, Janardhan Rao Doppa, and Alan Fern. 2020. Optimizing discrete spaces via expensive evaluations: A learning to search framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 3773–3780.

[38] Josip Djolonga, Andreas Krause, and Volkan Cevher. 2013. High-dimensional Gaussian process bandits. In *Neural Information Processing Systems*.

[39] David K Duvenaud, Hannes Nickisch, and Carl Rasmussen. 2011. Additive Gaussian processes. *Advances in neural information processing systems* 24 (2011).

[40] Cynthia Dwork. 2008. Differential Privacy: A Survey of Results. In *TAMC 2008*. 1–19.

[41] Michael TM Emmerich, André H Deutz, and Jan Willem Klinkenberg. 2011. Hypervolume-based expected improvement: Monotonicity properties and exact computation. In *2011 IEEE Congress of Evolutionary Computation (CEC)*. IEEE, 2147–2154.

[42] Michael TM Emmerich, Kyriakos C Giannakoglou, and Boris Naujoks. 2006. Single-and multiobjective evolutionary optimization assisted by Gaussian random field metamodels. *IEEE Transactions on Evolutionary Computation* 10, 4 (2006), 421–439.

[43] David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. 2019. Scalable global optimization via local Bayesian optimization. *Advances in Neural Information Processing Systems* 32 (2019), 5496–5507.

[44] Matthias Feurer, Benjamin Letham, and Eytan Bakshy. 2018. Scalable meta-learning for Bayesian optimization. *Stat* 1050 (2018), 6.

[45] Matthias Feurer, Jost Springenberg, and Frank Hutter. 2015. Initializing Bayesian hyperparameter optimization via meta-learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[46] Alexander IJ Forrester, Andy J Keane, and Neil W Bressloff. 2006. Design and analysis of "noisy" computer experiments. *AIAA journal* 44, 10 (2006), 2331–2339.

[47] Peter I Frazier. 2018. A Tutorial on Bayesian Optimization. *stat* 1050 (2018), 8.

[48] Peter I Frazier, Warren B Powell, and Savas Dayanik. 2008. A knowledge-gradient policy for sequential information collection. *SIAM Journal on Control and Optimization* 47, 5 (2008), 2410–2439.

[49] J Garcia-Barcos and R Martinez-Cantin. 2019. Fully distributed Bayesian optimization with stochastic policies. In *IJCAI International Joint Conference on Artificial Intelligence*.

[50] Jacob Gardner, Chuan Guo, Kilian Weinberger, Roman Garnett, and Roger Grosse. 2017. Discovering and exploiting additive structure for Bayesian optimization. In *Artificial Intelligence and Statistics*. PMLR, 1311–1319.

[51] Jacob R Gardner, Matt J Kusner, Zhixiang Eddie Xu, Kilian Q Weinberger, and John P Cunningham. 2014. Bayesian Optimization with Inequality Constraints.. In *ICML*, Vol. 2014. 937–945.

[52] Eduardo C Garrido-Merchán and Daniel Hernández-Lobato. 2020. Dealing with categorical and integer-valued variables in Bayesian optimization with Gaussian processes. *Neurocomputing* 380 (2020), 20–35.

[53] David Ginsbourger, Rodolphe Le Riche, and Laurent Carraro. 2008. A multi-points criterion for deterministic parallel global optimization based on Gaussian processes. (2008).

[54] David Ginsbourger, Rodolphe Le Riche, and Laurent Carraro. 2010. Kriging is well-suited to parallelize optimization. In *Computational Intelligence in Expensive Optimization Problems*. Springer, 131–162.

[55] Paul W Goldberg, Christopher KI Williams, and Christopher M Bishop. 1997. Regression with input-dependent noise: A Gaussian process treatment. *Advances in Neural Information Processing Systems* 10 (1997), 493–499.

[56] Daniel Golovin, Benjamin Solnik, Subhodeep Moitra, Greg Kochanski, John Karro, and David Sculley. 2017. Google vizier: A service for black-box optimization. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1487–1495.

[57] Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. 2018. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science* 4, 2 (2018), 268–276.

[58] Chengyue Gong, Jian Peng, and Qiang Liu. 2019. Quantile stein variational gradient descent for batch Bayesian optimization. In *International Conference on Machine Learning*. PMLR, 2347–2356.

[59] Javier González, Zhenwen Dai, Philipp Hennig, and Neil Lawrence. 2016. Batch Bayesian optimization via local penalization. In *Artificial Intelligence and Statistics*. PMLR, 648–657.

[60] Robert B Gramacy, Genetha A Gray, Sébastien Le Digabel, Herbert KH Lee, Pritam Ranjan, Garth Wells, and Stefan M Wild. 2016. Modeling an augmented Lagrangian for blackbox constrained optimization. *Technometrics* 58, 1 (2016), 1–11.

[61] Alex Graves. 2011. Practical variational inference for neural networks. *Advances in Neural Information Processing Systems* 24 (2011).

[62] Dan Guo, Yaochu Jin, Jinliang Ding, and Tianyou Chai. 2019. Heterogeneous ensemble-based infill criterion for evolutionary multiobjective optimization of expensive problems. *IEEE Transactions on Cybernetics* 49, 3 (2019), 1012–1025.

[63] Dan Guo, Xilu Wang, Kailai Gao, Yaochu Jin, Jinliang Ding, and Tianyou Chai. 2021. Evolutionary optimization of high-dimensional multiobjective and many-objective expensive problems assisted by a dropout neural network. *IEEE Transactions on Systems, Man, and Cybernetics: systems* (2021).

[64] Sunil Gupta, Alistair Shilton, Santu Rana, and Svetha Venkatesh. 2018. Exploiting strategy-space diversity for batch Bayesian optimization. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 538–547.

[65] Ali Hebbal, Mathieu Balesdent, Loïc Brevault, Nouredine Melab, and El-Ghazali Talbi. 2022. Deep Gaussian process for multi-objective Bayesian optimization. *Optimization and Engineering* (2022), 1–40.

[66] Ali Hebbal, Loïc Brevault, Mathieu Balesdent, El-Ghazali Talbi, and Nouredine Melab. 2021. Bayesian optimization using deep Gaussian processes with applications to aerospace system design. *Optimization and Engineering* 22, 1 (2021), 321–361.

[67] Ali Hebbal, Loic Brevault, Mathieu Balesdent, El-Ghazali Talbi, and Nouredine Melab. 2021. Multi-fidelity modeling with different input domain definitions using Deep Gaussian Processes. *Structural and Multidisciplinary Optimization* 63, 5 (2021), 2267–2288.

[68] Philipp Hennig and Christian J Schuler. 2012. Entropy Search for Information-Efficient Global Optimization. *Journal of Machine Learning Research* 13, 6 (2012).

[69] Daniel Hernández-Lobato, Jose Hernandez-Lobato, Amar Shah, and Ryan Adams. 2016. Predictive entropy search for multi-objective Bayesian optimization. In *International Conference on Machine Learning*. PMLR, 1492–1501.

[70] José Miguel Hernández-Lobato and Ryan Adams. 2015. Probabilistic backpropagation for scalable learning of Bayesian neural networks. In *International Conference on Machine Learning*. PMLR, 1861–1869.

[71] José Miguel Hernández-Lobato, Michael Gelbart, Matthew Hoffman, Ryan Adams, and Zoubin Ghahramani. 2015. Predictive entropy search for Bayesian optimization with unknown constraints. In *International Conference on Machine Learning*. PMLR, 1699–1707.

[72] José Miguel Hernández-Lobato, Michael A Gelbart, Ryan P Adams, Matthew W Hoffman, and Zoubin Ghahramani. 2016. A general framework for constrained Bayesian optimization using information-based search. *Journal of Machine Learning Research* (2016).

[73] José Miguel Hernández-Lobato, Matthew W Hoffman, and Zoubin Ghahramani. 2014. Predictive entropy search for efficient global optimization of black-box functions. *Advances in Neural Information Processing Systems* 27 (2014).

[74] José Miguel Hernández-Lobato, Edward Pyzer-Knapp, Alan Aspuru-Guzik, and Ryan P Adams. 2016. Distributed Thompson sampling for large-scale accelerated exploration of chemical space. In *NIPS Workshop on Bayesian Optimization*.

[75] José Miguel Hernández-Lobato, James Requeima, Edward O Pyzer-Knapp, and Alán Aspuru-Guzik. 2017. Parallel and distributed Thompson sampling for large-scale accelerated exploration of chemical space. In *International Conference on Machine Learning*. PMLR, 1470–1479.

[76] Trong Nghia Hoang, Quang Minh Hoang, Ruofei Ouyang, and Kian Hsiang Low. 2018. Decentralized high-dimensional Bayesian optimization with factor graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[77] Hanbin Hu, Peng Li, and Jianhua Z Huang. 2018. Parallelizable Bayesian optimization for analog and mixed-signal rare failure detection with high coverage. In *Proceedings of the International Conference on Computer-Aided Design*. 1–8.

[78] Deng Huang, Theodore T Allen, William I Notz, and R Allen Miller. 2006. Sequential kriging optimization using multiple-fidelity evaluations. *Structural and Multidisciplinary Optimization* 32, 5 (2006), 369–382.

[79] Deng Huang, Theodore T Allen, William I Notz, and Ning Zeng. 2006. Global optimization of stochastic black-box systems via sequential kriging meta-models. *Journal of Global Optimization* 34, 3 (2006), 441–466.

[80] Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. 2010. Sequential model-based optimization for general algorithm configuration (extended version). *Technical Report TR-2010–10, University of British Columbia, Computer Science, Tech. Rep.* (2010).

[81] Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. 2011. Sequential model-based optimization for general algorithm configuration. In *International Conference on Learning and Intelligent Optimization*. Springer, 507–523.

[82] Janis Janusevskis, Rodolphe Le Riche, David Ginsbourger, and Ramunas Girdziusas. 2012. Expected improvements for the asynchronous parallel global optimization of expensive functions: Potentials and challenges. In *International Conference on Learning and Intelligent Optimization*. Springer, 413–418.

[83] Noémie Jaquier, Leonel Rozo, Sylvain Calinon, and Mathias Bürger. 2019. Bayesian optimization meets Riemannian manifolds in robot learning. In *Conference on Robot Learning*. PMLR, 233–246.

[84] Shinkyu Jeong and Shigeru Obayashi. 2005. Efficient global optimization (EGO) for multi-objective problem and data mining. In *2005 IEEE Congress on Evolutionary Computation*, Vol. 3. IEEE, 2138–2145.

[85] Ruwang Jiao, Sanyou Zeng, Changhe Li, Yuhong Jiang, and Yaochu Jin. 2019. A complete expected improvement criterion for Gaussian process assisted highly constrained expensive optimization. *Information Sciences* 471 (2019), 80–96.

[86] Donald R Jones, Matthias Schonlau, and William J Welch. 1998. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization* 13, 4 (1998), 455–492.

[87] Tinu Theckel Joy, Santu Rana, Sunil Gupta, and Svetha Venkatesh. 2019. A flexible transfer learning framework for Bayesian optimization with convergence guarantee. *Expert Systems with Applications* 115 (2019), 656–672.

[88] Tinu Theckel Joy, Santu Rana, Sunil Gupta, and Svetha Venkatesh. 2020. Batch Bayesian optimization using multi-scale search. *Knowledge-Based Systems* 187 (2020), 104818.

[89] Kirthevasan Kandasamy, Gautam Dasarathy, Junier Oliva, Jeff Schneider, and Barnabás Póczos. 2016. Gaussian process optimisation with multi-fidelity evaluations. In *Proceedings of the 30th/International Conference on Advances in Neural Information Processing Systems (NIPS'30)*.

[90] Kirthevasan Kandasamy, Gautam Dasarathy, Junier Oliva, Jeff Schneider, and Barnabas Poczos. 2019. Multi-fidelity Gaussian process bandit optimisation. *Journal of Artificial Intelligence Research* 66 (2019), 151–196.

[91] Kirthevasan Kandasamy, Gautam Dasarathy, Jeff Schneider, and Barnabás Póczos. 2017. Multi-fidelity Bayesian optimisation with continuous approximations. In *International Conference on Machine Learning*. PMLR, 1799–1808.

[92] Kirthevasan Kandasamy, Akshay Krishnamurthy, Jeff Schneider, and Barnabás Póczos. 2018. Parallelised bayesian optimisation via thompson sampling. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 133–142.

[93] Kirthevasan Kandasamy, Willie Neiswanger, Jeff Schneider, Barnabas Poczos, and Eric Xing. 2018. Neural architecture search with Bayesian optimisation and optimal transport. *Advances in Neural Information Processing Systems* 31 (2018).

[94] Kirthevasan Kandasamy, Jeff Schneider, and Barnabás Póczos. 2015. High dimensional Bayesian optimisation and bandits via additive models. In *International Conference on Machine Learning*. PMLR, 295–304.

[95] Andy J Keane. 2006. Statistical improvement criteria for use in multiobjective design optimization. *AIAA journal* 44, 4 (2006), 879–891.

[96] Marc C Kennedy and Anthony O'Hagan. 2000. Predicting the output from a complex computer code when fast approximations are available. *Biometrika* 87, 1 (2000), 1–13.

[97] Jack PC Kleijnen. 2009. Kriging metamodeling in simulation: A review. *European Journal of Operational Research* 192, 3 (2009), 707–716.

[98] Joshua Knowles. 2006. ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation* 10, 1 (2006), 50–66.

[99] Harold J Kushner. 1964. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *Journal of Basic Engineering* 86, 1 (1964), 97–106.

[100] Remi Lam and Karen Willcox. 2017. Lookahead Bayesian Optimization with Inequality Constraints.. In *NIPS*. 1890–1900.

[101] Remi Lam, Karen Willcox, and David H Wolpert. 2016. Bayesian optimization with a finite budget: An approximate dynamic programming approach. *Advances in Neural Information Processing Systems* 29 (2016), 883–891.

[102] Miguel Lázaro-Gredilla and Michalis K Titsias. 2011. Variational heteroscedastic Gaussian process regression. In *ICML*.

[103] Loic Le Gratiet and Josselin Garnier. 2014. Recursive co-kriging model for design of computer experiments with multiple levels of fidelity. *International Journal for Uncertainty Quantification* 4, 5 (2014).

[104] Eric Hans Lee, David Eriksson, Valerio Perrone, and Matthias Seeger. 2021. A Nonmyopic Approach to Cost-Constrained Bayesian Optimization. In *Uncertainty in Artificial Intelligence*. PMLR, 568–577.

[105] Benjamin Letham, Roberto Calandra, Akshara Rai, and Eytan Bakshy. 2020. Re-examining linear embeddings for high-dimensional Bayesian optimization. *arXiv preprint arXiv:2001.11659* (2020).

[106] Benjamin Letham, Brian Karrer, Guilherme Ottoni, and Eytan Bakshy. 2019. Constrained Bayesian optimization with noisy experiments. *Bayesian Analysis* 14, 2 (2019), 495–519.

[107] B. Li, J. Li, K. Tang, and X. Yao. 2015. Many-Objective Evolutionary Algorithms: A Survey. *AcM Computing Surveys* 48, 1 (2015), Article No.: 13, pp 1–35.

[108] Chun-Liang Li, Kirthevasan Kandasamy, Barnabás Póczos, and Jeff Schneider. 2016. High dimensional Bayesian optimization via restricted projection pursuit models. In *Artificial Intelligence and Statistics*. PMLR, 884–892.

[109] Nan Li, Lin Yang, Xiaodong Li, Xiangdong Li, Jiyuan Tu, and Sherman CP Cheung. 2019. Multi-objective optimization for designing of high-speed train cabin ventilation system using particle swarm optimization and multi-fidelity Kriging. *Building and Environment* 155 (2019), 161–174.

[110] Zheng Li, Xinyu Wang, Shilun Ruan, Zhaojun Li, Changyu Shen, and Yan Zeng. 2018. A modified hypervolume based expected improvement for multi-objective efficient global optimization method. *Structural and Multidisciplinary Optimization* 58, 5 (2018), 1961–1979.

[111] Haitao Liu, Jianfei Cai, and Yew-Soon Ong. 2018. Remarks on multi-output Gaussian process regression. *Knowledge-Based Systems* 144 (2018), 102–121.

[112] Haitao Liu, Yew-Soon Ong, Xiaobo Shen, and Jianfei Cai. 2020. When Gaussian process meets big data: A review of scalable GPs. *IEEE Transactions on Neural Networks and Learning Systems* 31, 11 (2020), 4405–4423.

[113] Jingfei Liu, Chao Jiang, and Jing Zheng. 2021. Batch Bayesian optimization via adaptive local search. *Applied Intelligence* 51, 3 (2021), 1280–1295.

[114] Yixin Liu, Shishi Chen, Fenggang Wang, and Fenfen Xiong. 2018. Sequential optimization using multi-level cokriging and extended expected improvement criterion. *Structural and Multidisciplinary Optimization* 58, 3 (2018), 1155–1173.

[115] Zhiming Lv, Linqing Wang, Zhongyang Han, Jun Zhao, and Wei Wang. 2019. Surrogate-assisted particle swarm optimization algorithm with Pareto active learning for expensive multi-objective optimization. *IEEE/CAA Journal of Automatica Sinica* 6, 3 (2019), 838–849.

[116] Wesley J Maddox, Maximilian Balandat, Andrew G Wilson, and Eytan Bakshy. 2021. Bayesian optimization with high-dimensional outputs. *Advances in Neural Information Processing Systems* 34 (2021), 19274–19287.

[117] Alonso Marco, Felix Berkenkamp, Philipp Hennig, Angela P Schoellig, Andreas Krause, Stefan Schaal, and Sebastian Trimpe. 2017. Virtual vs. real: Trading off simulations and physical experiments in reinforcement learning with Bayesian optimization. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1557–1563.

[118] Ruben Martinez-Cantin, Kevin Tee, and Michael McCourt. 2018. Practical Bayesian optimization in the presence of outliers. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1722–1731.

[119] Julien Marzat, Eric Walter, and Hélène Piet-Lahanier. 2013. Worst-case global optimization of black-box functions through Kriging and relaxation. *Journal of Global Optimization* 55, 4 (2013), 707–727.

[120] Andrew McHutchon and Carl Rasmussen. 2011. Gaussian process training with input noise. *Advances in Neural Information Processing Systems* 24 (2011), 1341–1349.

[121] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*. PMLR, 1273–1282.

[122] Alan Tan Wei Min, Abhishek Gupta, and Yew-Soon Ong. 2020. Generalizing transfer Bayesian optimization to source-target heterogeneity. *IEEE Transactions on Automation Science and Engineering* (2020).

[123] Alan Tan Wei Min, Yew-Soon Ong, Abhishek Gupta, and Chi-Keong Goh. 2017. Multiproblem surrogates: Transfer evolutionary multiobjective optimization of computationally expensive problems. *IEEE Transactions on Evolutionary Computation* 23, 1 (2017), 15–28.

[124] Jonas Močkus. 1975. On Bayesian methods for seeking the extremum. In *Optimization Techniques IFIP Technical Conference*. Springer, 400–404.

[125] Riccardo Moriconi, Marc Peter Deisenroth, and KS Sesh Kumar. 2020. High-dimensional Bayesian optimization using low-dimensional feature spaces. *Machine Learning* 109, 9 (2020), 1925–1943.

[126] Henry B Moss, David S Leslie, and Paul Rayson. 2020. Mumbo: Multi-task max-value Bayesian optimization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 447–462.

[127] Mojmír Mutnỳ and Andreas Krause. 2018. Efficient high dimensional Bayesian optimization with additivity and quadrature fourier features. *Advances in Neural Information Processing Systems 31* (2018), 9005–9016.

[128] Donald E Myers. 1982. Matrix formulation of co-kriging. *Journal of the International Association for Mathematical Geology* 14, 3 (1982), 249–257.

[129] Amin Nayebi, Alexander Munteanu, and Matthias Poloczek. 2019. A framework for Bayesian optimization in embedded subspaces. In *International Conference on Machine Learning*. PMLR, 4752–4761.

[130] Dang Nguyen, Sunil Gupta, Santu Rana, Alistair Shilton, and Svetha Venkatesh. 2020. Bayesian optimization for categorical and category-specific continuous inputs. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 5256–5263.

[131] Quoc Phong Nguyen, Zhaoxuan Wu, Bryan Kian Hsiang Low, and Patrick Jaillet. 2021. Trusted-maximizers entropy search for efficient Bayesian optimization. In *Uncertainty in Artificial Intelligence*. PMLR, 1486–1495.

[132] Vu Nguyen, Tam Le, Makoto Yamada, and Michael A Osborne. 2021. Optimal transport kernels for sequential and parallel neural architecture search. In *International Conference on Machine Learning*. PMLR, 8084–8095.

[133] José Nogueira, Ruben Martinez-Cantin, Alexandre Bernardino, and Lorenzo Jamone. 2016. Unscented Bayesian optimization for safe robot grasping. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 1967–1972.

[134] ChangYong Oh, Efstratios Gavves, and Max Welling. 2018. BOCK: Bayesian optimization with cylindrical kernels. In *International Conference on Machine Learning*. PMLR, 3868–3877.

[135] Changyong Oh, Jakub Tomczak, Efstratios Gavves, and Max Welling. 2019. Combinatorial Bayesian optimization using the graph cartesian product. *Advances in Neural Information Processing Systems* 32 (2019).

[136] Anthony O'Hagan. 1979. On outlier rejection phenomena in Bayes inference. *Journal of the Royal Statistical Society: Series B (Methodological)* 41, 3 (1979), 358–367.

[137] Christopher J Paciorek and Mark J Schervish. 2006. Spatial modelling using a new class of nonstationary covariance functions. *Environmetrics: The Official Gournal of the International Environmetrics Society* 17, 5 (2006), 483–506.

[138] Julien Pelamatti, Loïc Brevault, Mathieu Balesdent, El-Ghazali Talbi, and Yannick Guerin. 2019. Efficient global optimization of constrained mixed variable problems. *Journal of Global Optimization* 73, 3 (2019), 583–613.

[139] Julien Pelamatti, Loïc Brevault, Mathieu Balesdent, El-Ghazali Talbi, and Yannick Guerin. 2021. Bayesian optimization of variable-size design space problems. *Optimization and Engineering* 22, 1 (2021), 387–447.

[140] Paris Perdikaris, Maziar Raissi, Andreas Damianou, Neil D Lawrence, and George Em Karniadakis. 2017. Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 473, 2198 (2017), 20160751.

[141] Valerio Perrone, Michele Donini, Muhammad Bilal Zafar, Robin Schmucker, Krishnaram Kenthapadi, and Cédric Archambeau. 2021. Fair Bayesian optimization. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 854–863.

[142] Valerio Perrone, Iaroslav Shcherbatyi, Rodolphe Jenatton, Cedric Archambeau, and Matthias Seeger. 2019. Constrained Bayesian optimization with max-value entropy search. *33rd Conference on Neural Information Processing Systems* (2019).

[143] Victor Picheny. 2014. A stepwise uncertainty reduction approach to constrained global optimization. In *Artificial Intelligence and Statistics*. PMLR, 787–795.

[144] Victor Picheny, David Ginsbourger, Yann Richet, and Gregory Caplin. 2013. Quantile-based optimization of noisy computer experiments with tunable precision. *Technometrics* 55, 1 (2013), 2–13.

[145] Victor Picheny, Tobias Wagner, and David Ginsbourger. 2013. A benchmark of kriging-based infill criteria for noisy optimization. *Structural and Multidisciplinary Optimization* 48, 3 (2013), 607–626.

[146] Wolfgang Ponweiser, Tobias Wagner, Dirk Biermann, and Markus Vincze. 2008. Multiobjective optimization on a limited budget of evaluations using model-assisted $\mathcal{S}$-metric selection. In *International Conference on Parallel Problem Solving from Nature*. Springer, 784–794.

[147] Peter ZG Qian and CF Jeff Wu. 2008. Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments. *Technometrics* 50, 2 (2008), 192–204.

[148] Shufen Qin, Chaoli Sun, Yaochu Jin, and Guochen Zhang. 2019. Bayesian approaches to surrogate-assisted evolutionary multi-objective optimization: a comparative study. In *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2074–2080.

[149] Anil Ramachandran, Sunil Gupta, Santu Rana, and Svetha Venkatesh. 2018. Selecting optimal source for transfer learning in Bayesian optimisation. In *Pacific Rim International Conference on Artificial Intelligence*. Springer, 42–56.

[150] Santu Rana, Cheng Li, Sunil Gupta, Vu Nguyen, and Svetha Venkatesh. 2017. High dimensional Bayesian optimization with elastic Gaussian process. In *International Conference on Machine Learning*. PMLR, 2883–2891.

[151] Carl Rasmussen and Zoubin Ghahramani. 2001. Infinite mixtures of Gaussian process experts. *Advances in Neural Information Processing Systems* 14 (2001).

[152] Carl Edward Rasmussen. 2003. Gaussian processes in machine learning. In *Summer School on Machine Learning*. Springer, 63–71.

[153] Paul Rolland, Jonathan Scarlett, Ilija Bogunovic, and Volkan Cevher. 2018. High-dimensional Bayesian optimization via additive models with overlapping groups. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 298–307.

[154] Binxin Ru, Ahsan Alvi, Vu Nguyen, Michael A Osborne, and Stephen Roberts. 2020. Bayesian optimisation over multiple continuous and categorical inputs. In *International Conference on Machine Learning*. PMLR, 8276–8285.

[155] Binxin Ru, Xingchen Wan, Xiaowen Dong, and Michael Osborne. 2021. Interpretable Neural Architecture Search via Bayesian Optimisation with Weisfeiler-Lehman Kernels. In *International Conference on Learning Representations*.

[156] Xiaoran Ruan, Ke Li, Bilel Derbel, and Arnaud Liefooghe. 2020. Surrogate assisted evolutionary algorithm for medium scale multi-objective optimisation problems. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*. 560–568.

[157] Jerome Sacks, William J Welch, Toby J Mitchell, and Henry P Wynn. 1989. Design and analysis of computer experiments. *Statist. Sci.* 4, 4 (1989), 409–423.

[158] Michael James Sasena. 2002. *Flexibility and efficiency enhancements for constrained global design optimization with kriging approximations.* University of Michigan.

[159] Nicolas Schilling, Martin Wistuba, and Lars Schmidt-Thieme. 2016. Scalable hyperparameter optimization with products of Gaussian process experts. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 33–48.

[160] Matthias Schonlau, William J Welch, and Donald R Jones. 1998. Global versus local search in constrained optimization of computer models. *Lecture Notes-Monograph Series* (1998), 11–25.

[161] Warren Scott, Peter Frazier, and Warren Powell. 2011. The correlated knowledge gradient for simulation optimization of continuous parameters using Gaussian process regression. *SIAM Journal on Optimization* 21, 3 (2011), 996–1026.

[162] Rajat Sen, Kirthevasan Kandasamy, and Sanjay Shakkottai. 2018. Multi-fidelity black-box optimization with hierarchical partitions. In *International Conference on Machine Learning*. PMLR, 4538–4547.

[163] Amar Shah and Zoubin Ghahramani. 2015. Parallel predictive entropy search for batch global optimization of expensive objective functions. *arXiv preprint arXiv:1511.07130* (2015).

[164] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. 2016. Taking the human out of the loop: A review of Bayesian optimization. *Proc. IEEE* 104, 1 (2016), 148–175.

[165] Koji Shimoyama, Koma Sato, Shinkyu Jeong, and Shigeru Obayashi. 2012. Comparison of the criteria for updating kriging response surface models in multi-objective optimization. In *2012 IEEE Congress on Evolutionary Computation*. IEEE, 1–8.

[166] Eero Siivola, Andrei Paleyes, Javier González, and Aki Vehtari. 2021. Good practices for Bayesian optimization of high dimensional structured spaces. *Applied AI Letters* 2, 2 (2021), e24.

[167] Rachael Hwee Ling Sim, Yehong Zhang, Bryan Kian Hsiang Low, and Patrick Jaillet. 2021. Collaborative Bayesian optimization with fair regret. In *International Conference on Machine Learning*. PMLR, 9691–9701.

[168] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. 2012. Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems* 25 (2012).

[169] Jasper Snoek, Oren Rippel, Kevin Swersky, Ryan Kiros, Nadathur Satish, Narayanan Sundaram, Mostofa Patwary, Mr Prabhat, and Ryan Adams. 2015. Scalable Bayesian optimization using deep neural networks. In *International Conference on Machine Learning*. PMLR, 2171–2180.

[170] Jasper Snoek, Kevin Swersky, Rich Zemel, and Ryan Adams. 2014. Input warping for Bayesian optimization of non-stationary functions. In *International Conference on Machine Learning*. PMLR, 1674–1682.

[171] Adrien Spagnol, Rodolphe Le Riche, and Seébastien Da Veiga. 2019. Global sensitivity analysis for optimization with variable selection. *SIAM/ASA Journal on Uncertainty Quantification* 7, 2 (2019), 417–443.

[172] Jost Tobias Springenberg, Aaron Klein, Stefan Falkner, and Frank Hutter. 2016. Bayesian optimization with robust Bayesian neural networks. *Advances in Neural Information Processing Systems* 29 (2016), 4134–4142.

[173] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. 2010. Gaussian process optimization in the bandit setting: No regret and experimental design. *Proceedings of the 27 th International Conference on Machine Learning* (2010).

[174] Shinya Suzuki, Shion Takeno, Tomoyuki Tamura, Kazuki Shitara, and Masayuki Karasuyama. 2020. Multi-objective Bayesian optimization using Pareto-frontier entropy. In *International Conference on Machine Learning*. PMLR, 9279–9288.

[175] Joshua Svenson and Thomas Santner. 2016. Multiobjective optimization of expensive-to-evaluate deterministic computer simulator models. *Computational Statistics & Data Analysis* 94 (2016), 250–264.

[176] Kevin Swersky, Yulia Rubanova, David Dohan, and Kevin Murphy. 2020. Amortized Bayesian optimization over discrete spaces. In *Conference on Uncertainty in Artificial Intelligence*. PMLR, 769–778.

[177] Kevin Swersky, Jasper Snoek, and Ryan Prescott Adams. 2013. Multi-task Bayesian optimization. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc.

[178] Shion Takeno, Hitoshi Fukuoka, Yuhki Tsukada, Toshiyuki Koyama, Motoki Shiga, Ichiro Takeuchi, and Masayuki Karasuyama. 2020. Multi-fidelity Bayesian optimization with max-value entropy search and its parallelization. In *International Conference on Machine Learning*. PMLR, 9334–9345.

[179] Matteo Turchetta, Andreas Krause, and Sebastian Trimpe. 2020. Robust model-free reinforcement learning with multi-objective Bayesian optimization. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 10702–10708.

[180] Samee ur Rehman, Matthijs Langelaar, and Fred van Keulen. 2014. Efficient Kriging-based robust optimization of unconstrained problems. *Journal of Computational Science* 5, 6 (2014), 872–881.

[181] Wim CM Van Beers and Jack PC Kleijnen. 2003. Kriging for interpolation in random simulation. *Journal of the Operational Research Society* 54, 3 (2003), 255–262.

[182] Jarno Vanhatalo, Pasi Jylänki, and Aki Vehtari. 2009. Gaussian process regression with Student-t likelihood. *Advances in Neural Information Processing Systems* 22 (2009), 1910–1918.

[183] Michael Volpp, Lukas P. Fröhlich, Kirsten Fischer, Andreas Doerr, Stefan Falkner, Frank Hutter, and Christian Daniel. 2020. Meta-Learning Acquisition Functions for Transfer Learning in Bayesian Optimization. In *International Conference on Learning Representations*.

[184] Tobias Wagner, Michael Emmerich, André Deutz, and Wolfgang Ponweiser. 2010. On expected-improvement criteria for model-based multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature*. Springer, 718–727.

[185] Jialei Wang, Scott C Clark, Eric Liu, and Peter I Frazier. 2020. Parallel Bayesian global optimization of expensive functions. *Operations Research* 68, 6 (2020), 1850–1865.

[186] Xilu Wang, Yaochu Jin, Sebastian Schmitt, and Markus Olhofer. 2020. An adaptive Bayesian approach to surrogate-assisted evolutionary multi-objective optimization. *Information Sciences* 519 (2020), 317–331.

[187] Xilu Wang, Yaochu Jin, Sebastian Schmitt, and Markus Olhofer. 2022. Alleviating Search Bias in Bayesian Evolutionary Optimization with Heterogeneous Objectives. (2022). Manuscript submitted for publication.

[188] Xilu Wang, Yaochu Jin, Sebastian Schmitt, and Markus Olhofer. 2022. Transfer Learning Based Co-surrogate Assisted Evolutionary Bi-objective Optimization for Objectives with Non-uniform Evaluation Times. *Evolutionary computation* (2022), 1–27.

[189] Xilu Wang, Yaochu Jin, Sebastian Schmitt, Markus Olhofer, and Richard Allmendinger. 2021. Transfer learning based surrogate assisted evolutionary bi-objective optimization for objectives with different evaluation times. *Knowledge-Based Systems* (2021), 107190.

[190] Zi Wang, Clement Gehring, Pushmeet Kohli, and Stefanie Jegelka. 2018. Batched large-scale Bayesian optimization in high-dimensional spaces. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 745–754.

[191] Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Feitas. 2016. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research* 55 (2016), 361–387.

[192] Zi Wang and Stefanie Jegelka. 2017. Max-value entropy search for efficient Bayesian optimization. In *International Conference on Machine Learning*. PMLR, 3627–3635.

[193] Zi Wang, Chengtao Li, Stefanie Jegelka, and Pushmeet Kohli. 2017. Batched high-dimensional Bayesian optimization via structural kernel learning. In *International Conference on Machine Learning*. PMLR, 3656–3664.

[194] Ziyu Wang, Masrour Zoghi, Frank Hutter, David Matheson, and Nando De Freitas. 2013. Bayesian optimization in high dimensions via random embeddings. In *Twenty-Third International Joint Conference on Artificial Intelligence*.

[195] Chris Williams, Edwin V Bonilla, and Kian M Chai. 2007. Multi-task Gaussian process prediction. *Advances in Neural Information Processing Systems* (2007), 153–160.

[196] Munir A Winkel, Jonathan W Stallrich, Curtis B Storlie, and Brian J Reich. 2021. Sequential Optimization in Locally Important Dimensions. *Technometrics* 63, 2 (2021), 236–248.

[197] Martin Wistuba, Nicolas Schilling, and Lars Schmidt-Thieme. 2018. Scalable Gaussian process-based transfer surrogates for hyperparameter optimization. *Machine Learning* 107, 1 (2018), 43–78.

[198] Jian Wu and Peter Frazier. 2016. The parallel knowledge gradient method for batch Bayesian optimization. *Advances in Neural Information Processing Systems* 29 (2016), 3126–3134.

[199] Hang Xu, Wenhua Zeng, Xiangxiang Zeng, and Gary G Yen. 2020. A polar-metric-based evolutionary algorithm. *IEEE Transactions on Cybernetics* (2020).

[200] Jinjin Xu, Yaochu Jin, and Wenli Du. 2021. A federated data-driven evolutionary algorithm for expensive multi-/many-objective optimization. *Complex & Intelligent Systems* 7, 6 (2021), 3093–3109.

[201] Jinjin Xu, Yaochu Jin, Wenli Du, and Sai Gu. 2021. A federated data-driven evolutionary algorithm. *Knowledge-Based Systems* 233 (2021), 107532.

[202] Kaifeng Yang, Michael Emmerich, André Deutz, and Thomas Bäck. 2019. Multi-objective Bayesian global optimization using expected hypervolume improvement gradient. *Swarm and Evolutionary Computation* 44 (2019), 945–956.

[203] Danial Yazdani, Ran Cheng, Donya Yazdani, Juergen Branke, Yaochu Jin, , and Xin Yao. 2021. A survey of evolutionary continuous dynamic optimization over two decades – Part A. *IEEE Transactions on Evolutionary Computation* 25, 4 (2021), 609–629.

[204] Dani Yogatama and Gideon Mann. 2014. Efficient transfer learning method for automatic hyperparameter tuning. In *Artificial Intelligence and Statistics*. PMLR, 1077–1085.

[205] M Todd Young, Jacob Hinkle, Arvind Ramanathan, and Ramakrishnan Kannan. 2018. Hyperspace: Distributed Bayesian hyperparameter optimization. In *2018 30th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*. IEEE, 339–347.

[206] Xubo Yue and Raed AL Kontar. 2020. Why non-myopic Bayesian optimization is promising and how far should we look-ahead? A study via rollout. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2808–2818.

[207] Dawei Zhan and Huanlai Xing. 2020. Expected improvement for expensive optimization: a review. *Journal of Global Optimization* 78, 3 (2020), 507–544.

[208] Miao Zhang, Huiqi Li, and Steven Su. 2019. High dimensional Bayesian optimization via supervised dimension reduction. *arXiv preprint arXiv:1907.08953* (2019).

[209] Qingfu Zhang and Hui Li. 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation* 11, 6 (2007), 712–731.

[210] Qingfu Zhang, Wudong Liu, Edward Tsang, and Botond Virginas. 2009. Expensive multiobjective optimization by MOEA/D with Gaussian process model. *IEEE Transactions on Evolutionary Computation* 14, 3 (2009), 456–474.

[211] Shuhan Zhang, Fan Yang, Changhao Yan, Dian Zhou, and Xuan Zeng. 2021. An Efficient Batch Constrained Bayesian Optimization Approach for Analog Circuit Synthesis via Multi-objective Acquisition Ensemble. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2021).

[212] Shuhan Zhang, Fan Yang, Changhao Yan, Dian Zhou, and Xuan Zeng. 2022. An Efficient Batch-Constrained Bayesian Optimization Approach for Analog Circuit Synthesis via Multiobjective Acquisition Ensemble. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* 41, 1 (2022), 1–14.

[213] Yehong Zhang, Trong Nghia Hoang, Bryan Kian Hsiang Low, and Mohan Kankanhalli. 2017. Information-based multi-fidelity Bayesian optimization. In *NIPS Workshop on Bayesian Optimization*.

[214] Yunxiang Zhang, Xiangyu Zhang, and Peter Frazier. 2021. Constrained Two-step Look-Ahead Bayesian Optimization. *Advances in Neural Information Processing Systems* 34 (2021).

[215] Peng Zhao, Lijun Zhang, Yuan Jiang, and Zhi-Hua Zhou. 2020. A simple approach for non-stationary linear bandits. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 746–755.

[216] Aimin Zhou, Yaochu Jin, Qingfu Zhang, Bernhard Sendhoff, and Edward Tsang. 2006. Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion. In *2006 IEEE International Conference on Evolutionary Computation*. IEEE, 892–899.

[217] A. Zhou, B. Qu, H. Li, S. Zhao, P. N. Suganthan, and Q. Zhang. 2011. Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation* 1, 1 (2011), 32–49.

[218] Xingyu Zhou and Ness Shroff. 2021. No-Regret Algorithms for Time-Varying Bayesian Optimization. In *2021 55th Annual Conference on Information Sciences and Systems (CISS)*. IEEE, 1–6.

[219] A Zilinskas et al. 1978. Optimization of one-dimensional multimodal functions. *Journal of the Royal Statistical Society, Series C (Applied Statistics)* 27, 3 (1978).

[220] Eckart Zitzler and Lothar Thiele. 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation* 3, 4 (1999), 257–271.

[221] Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M Fonseca, and Viviane Grunert Da Fonseca. 2003. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation* 7, 2 (2003), 117–132.