IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 27, 2019, accepted December 27, 2019, date of publication January 13, 2020, date of current version January 23, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2966228

Bayesian Optimization for Adaptive Experimental Design: A Review

STEWART GREENHILL, SANTU RANA, SUNIL GUPTA, PRATIBHA VELLANKI, AND SVETHA VENKATESH

Applied Artificial Intelligence Institute, Deakin University, Waurn Ponds Campus, Geelong, VIC 3216, Australia

Corresponding author: Stewart Greenhill (s.greenhill@deakin.edu.au)

This work was supported by the Australian Government through the Australian Research Council (ARC). The work of Svetha Venkatesh was supported by the ARC Australian Laureate Fellowship under Grant FL170100006.

ABSTRACT Bayesian optimisation is a statistical method that efficiently models and optimises expensive "black-box" functions. This review considers the application of Bayesian optimisation to experimental design, in comparison to existing Design of Experiments (DOE) methods. Solutions are surveyed for a range of core issues in experimental design including: the incorporation of prior knowledge, high dimensional optimisation, constraints, batch evaluation, multiple objectives, multi-fidelity data, and mixed variable types.

INDEX TERMS Bayesian methods, design for experiments, design optimization, machine learning algorithms.

I. INTRODUCTION

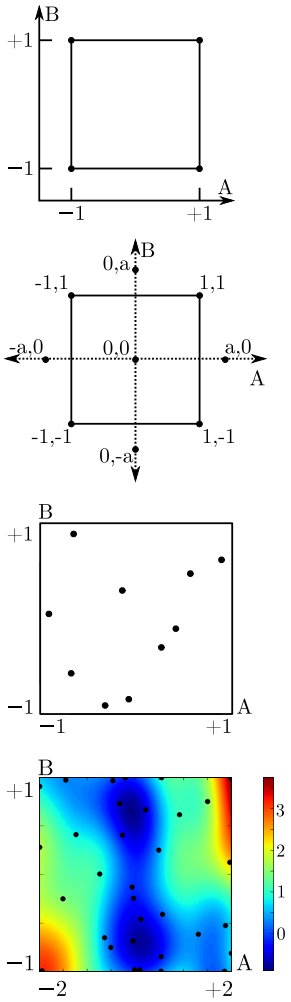
Experiments are fundamental to scientific and engineering practice. A well-designed experiment yields an empirical model of a process, which facilitates understanding and prediction of its behaviour. Experiments are often costly, so formal Design of Experiments methods (or DOE) [1]–[3] optimise measurement of the design space to give the best model from the fewest observations.

Models are important decision tools for design engineers. Understanding of design problems is enhanced when the design space can be explored cheaply and rapidly, allowing adjustment of the number and range of design variables, identification of ineffective constraints, balancing multiple design objectives, and optimisation [4]. Industrial processes must be robust to environmental conditions, component variation, and variability around a target [3]. Robust Parameter Design (RPD) [5]–[7] systematically characterises the influence of uncontrollable variables and noise. The number of observations required to build a model increases rapidly with the number of variables, making it challenging to investigate systems with many variables. Screening experiments can identify subsets of important variables to be later investigated in more detail [8], [9]. Optimisation is important in most industrial applications, and there are often multiple objectives which must balanced including yield, robustness, and cost. In classical experimental design, modelling and optimisation are separate processes, but newer model-based approaches can potentially sample more efficiently by adapting to the response surface, and can incorporate optimisation into the modelling process.

Machine learning has made great strides in the recent past, and we present here a machine learning approach to experimental design. **Bayesian Optimisation** (BO) [19], [20] is a powerful method for efficient global optimisation of expensive black-box functions. The experimental method introduces specific challenges: how to handle **constraints**, **high dimensionality**, **mixed variable types**, **multiple objectives**, **parallel (batch) evaluation**, and the transfer of **prior knowledge**. Several reviews have presented BO for a technical audience [20]–[22]. Our review surveys recent methods for systematically handling these challenges within a BO framework, with an emphasis on applications in science and engineering, and in the context of modern experimental design.

Bayesian optimisation is a **sample efficient** optimisation algorithm and thus suits optimisation of **expensive, black-box** systems. By "black-box" we mean that the objective function does not have a closed-form representation, does not provide function derivatives, and only allows point-wise evaluation. Several optimisation algorithms can handle optimisation of black-box functions such as

The associate editor coordinating the review of this manuscript and approving it for publication was Bijoy Chand Chatterjee.



Two-level Factorial Design

Factors are sampled at maxima and minima, which are coded -1 and 1. Two level designs estimate main effects and their products. Finer grids allow higher polynomial terms to be estimated. Fractional factorial designs estimate k main effects in $k+1$ samples, with some confounding between main effects and their products.

Model: linear

Samples: $N=2^k$

Central Composite Design

The addition of $2k+1$ axial samples to a factorial design allows the estimation of quadratic terms. These designs are used to estimate second order response surfaces for Response Surface Methodology (RSM)

Model: quadratic

Samples: $N=2^k+2k+1$

Space Filling Design

Distributes samples evenly throughout the space, so that each level of each factor is sampled with equal probability. Methods include latin hypercube sampling, uniform designs, sphere packing, and low-discrepancy sequences (eg. Hammersley, Sobol)

Model: independent

Samples: $N\approx 10k$ (heuristic)

Adaptive Sampling

Samples based on previous observations by balancing exploration (visiting unexplored areas) and exploitation (sampling in known interesting areas). May use model information (eg. Gaussian process variance), or be independent of the model type (eg. using local gradients or cross-validation). Figure colour shows mean of GP model of Camelback function.

Model: usually non-parametric, built incrementally and used to guide sampling

Samples: depends on model and required statistical significance of the estimate

FIGURE 1. Sampling methods used in experimental design. In classical Factorial designs samples are placed on a geometric grid. Space filling designs are used with a variety of non-linear models. Sample requirements are determined heuristically, but these designs are empirically much more efficient than grids.

multi-start derivative free local optimiser e.g. COBYLA [36], or evolutionary algorithms e.g. ISRES [37], or Lipschitzian methods such as DIRECT [34]. However, none of these are designed to be sample efficient, and all need to evaluate a function many times to perform optimisation. In contrast, Bayesian optimisation uses a model based approach with an adaptive sampling strategy to minimise the number of function evaluations.

Past approaches to experimental design have closely coupled sampling and modelling. Factorial designs assume a linear model and sample at orthogonal corners of the design space (see Figure 1). For more complex non-linear models, general purpose space-filling designs such as Latin hypercubes offer a more uniform coverage of the design space. For N sample points in k dimensions, there are $(N!)^{k-1}$ possible Latin hypercube designs, and finding a suitable design involves balancing space-filling (e.g. via entropy, or potential energy) with other desirable properties such as orthogonality. Much literature exists on the design of Latin hypercubes, and many research issues remain open [10], [11] such as: mixing of discrete and continuous variables, incorporation of global sensitivity information, and sequential sampling.

Response Surface Methodology (RSM) [3], [12] is a sequential approach which has become the primary method for industrial experimentation. In its original form, response surfaces are second order polynomials which are determined using central composite factorial experiments, and a path of steepest ascent is used to seek an optimal point. For robust design, replication is used to estimate noise factors, and optimisation must consider dual responses for process mean and variance. Approaches for handling multiple objectives include "split-plot" techniques, "desirability functions" and Pareto fronts [13]. Non-parametric RSM can be more general than second-order polynomials, and uses techniques such as Gaussian processes, thin-plate splines, and neural networks. Alternative optimisation approaches include simulated annealing, branch-and-bound and genetic algorithms [14].

In many areas, experiments are performed with detailed computer simulations of physical systems. Aerospace designers frequently work with expensive CFD (computational fluid dynamic) and FEA (finite element analysis) simulations. Multi-agent simulations are used to model how actor behaviour determines the outcome of group interactions in areas such as defence, networking, transportation, and logistics. Design and Analysis of Computer Experiments (or DACE, after [15]) differs from DOE in several ways. Simulations are generally deterministic, without random effects and uncontrolled variables, so less emphasis is placed on dealing with measurement noise. Simulations often include many variables, so there is more need to handle high dimensionality and mixed variable types. Where the response is complex, non-parametric models are used, including Gaussian Processes, Multivariate Adaptive Regression Splines, and Support Vector Regression [4], [16], [17].

A problem with classical DOE and space-filling designs is that the sampling pattern is determined before measurements are made, and cannot adapt to features that appear during the experiment. In contrast, **adaptive sampling** [16], [18] is a sequential process that decides the location of the next sample by balancing two criteria. Firstly, it samples in areas that have not been previously explored (e.g. based on distance from previous samples). Secondly, it samples more densely in areas where interesting behaviour is observed, such as rapid change or non-linearity. This can be detected using local gradients, prediction variance (e.g. where uncertainty is modelled), by checking agreement between the model and data (cross-validation), or agreement between an ensemble of models. BO is a form of model-based global optimisation (MBGO [16]), which uses adaptive sampling to guide the experiment towards a global optimum. Unlike pure adaptive sampling, MBGO considers the optimum of the modelled objective when deciding where to sample.

Recently, there has been a surge in applying Bayesian optimisation to design problems involving physical products and processes. In [23], Bayesian optimisation is applied in combination with a density functional theory (DFT) based computational tool to design low thermal hysteresis NiTi-based shape memory alloys. Similarly, in [24] Bayesian optimisation is used to optimise both the alloy composition and the associated heat treatment schedule to improve the performance of Al-7xxx series alloys. In [25], Bayesian optimisation is applied for high-quality nano-fibre design meeting a required specification of fibre length and diameter within few tens of iterations, greatly accelerating the production process. It has also been applied in other diverse fields including optimisation of nano-structures for optimal phonon transport [26], optimisation for maximum power point tracking in photovoltaic power plants [27], optimisation for efficient determination of metal oxide grain boundary structures [28], and for optimisation of computer game design to maximise engagement [29]. It has also been used in a recent neuroscience study [30] in designing cognitive tasks that maximally segregate ventral and dorsal FPN activity.

The recent advances in both the theory and practice of Bayesian optimisation has led to a plethora of techniques. In most parts, each advance is applicable to a sub-set of experimental conditions. What is lacking is both an overview of these methods and a methodology to adapt these techniques to a particular experimental design context. We fill this gap and provide a comprehensive study of the state-of-the-art Bayesian optimisation algorithms in terms of their applicability in experimental optimisation. Further, we provide a template of how disparate algorithms can be connected to create a fit-for-purpose solution. This thus provides an overview of the capability and increases the reach of these powerful methods. We conclude by discussion where further research is needed.

II. BAYESIAN OPTIMISATION

Bayesian optimisation incorporates two main ideas:

- A **Gaussian process** (GP) is used to maintain a belief over the design space. This simultaneously models the predicted mean $\mu_t(\boldsymbol{x})$ and the epistemic uncertainty $\sigma_t(\boldsymbol{x})$ at any point $\boldsymbol{x}$ in the input space, given a set of observations $\mathcal{D}_{1:t} = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), ...(\boldsymbol{x}_t, y_t)\}$, where $\boldsymbol{x}_t$ is the process input, and y_t is the corresponding output at time t.
- An **acquisition function** expresses the most promising setting for the next experiment, based on the predicted mean $\mu_t(\boldsymbol{x})$ and the uncertainty $\sigma_t(\boldsymbol{x})$.

A GP is completely specified by its mean function $m(\boldsymbol{x})$ and covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$:

$$f(\boldsymbol{x}) \sim \mathcal{GP}(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')) \tag{1}$$

The covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$ is also called the "kernel", and expresses the "smoothness" of the process. We expect that if two points $\boldsymbol{x}$ and $\boldsymbol{x}'$ are "close", then the corresponding process outputs y and y' will also be "close", and that the closeness depends on the distance between the points, and not the absolute location or direction of separation. A popular choice for the covariance function is the squared exponential (SE) function, also known as radial basis function (RBF):

$$k(\boldsymbol{x}, \boldsymbol{x}') = exp\left(-\frac{1}{2\theta^2}\left\|\boldsymbol{x} - \boldsymbol{x}'\right\|^2\right) \tag{2}$$

Equation 2 says that the correlation decreases with the square of the distance between points, and includes a parameter θ to define the length scale over which this happens. Specialised kernel functions are sometimes used to express pre-existing knowledge about the function (e.g. if something is known about the shape of f).

In an experimental setting, observations include a term for normally distributed noise $\epsilon \sim \mathcal{N}(0, \sigma^2_{noise})$, and the observation model is:

$$y = f(\boldsymbol{x}) + \epsilon$$

Gaussian process regression (or "kriging") can predict the value of the objective function $f(\cdot)$ at time $t + 1$ for any

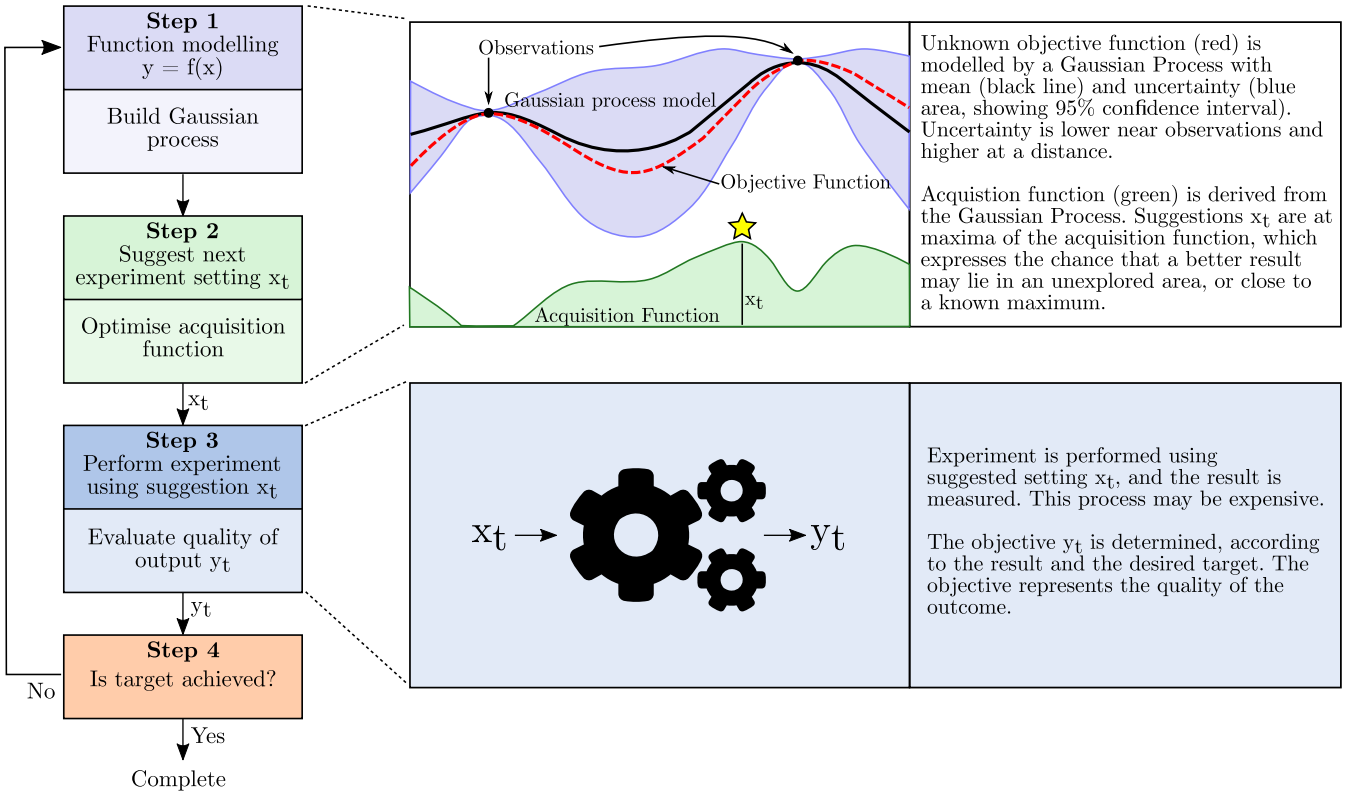


FIGURE 2. Bayesian optimisation is an iterative process in which the unknown system response is modelled using a Gaussian process. An acquisition function expresses the most promising setting for the next experiment, and can be efficiently optimised. The model quality improves progressively over time as successive measurements are incorporated.

location $\boldsymbol{x}$. The result is a normal distribution with mean $\mu_t(\boldsymbol{x})$ and uncertainty $\sigma_t(\boldsymbol{x})$.

$$P(f_{t+1} \mid \mathcal{D}_{1:t}, \boldsymbol{x}) = \mathcal{N}(\mu_t(\boldsymbol{x}), \sigma_t^2(\boldsymbol{x})) \tag{3}$$

where

$$\begin{aligned} \mu_t(\boldsymbol{x}) &= \boldsymbol{k}^T [K + \sigma_{noise}^2 I]^{-1} y_{1:t} \\ \sigma_t(\boldsymbol{x}) &= k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}^T [K + \sigma_{noise}^2 I]^{-1} \boldsymbol{k} \\ k &= [k(\boldsymbol{x}, \boldsymbol{x}_1), k(\boldsymbol{x}, \boldsymbol{x}_2), \ldots, k(\boldsymbol{x}, \boldsymbol{x}_t)] \end{aligned} \tag{4}$$

$$K = \begin{bmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & \ldots & k(\boldsymbol{x}_1, \boldsymbol{x}_t) \\ \vdots & \ddots & \vdots \\ k(\boldsymbol{x}_t, \boldsymbol{x}_1) & \ldots & k(\boldsymbol{x}_t, \boldsymbol{x}_t) \end{bmatrix} \tag{5}$$

Using the Gaussian process model, an **acquisition function** is constructed to represent the most promising setting for the next experiment. Acquisition functions are mainly derived from the $\mu(\boldsymbol{x})$ and $\sigma(\boldsymbol{x})$ of the GP model, and are hence cheap to compute. The acquisition function allows a balance between **exploitation** (sampling where the objective mean $\mu(\cdot)$ is high) and **exploration** (sampling where the uncertainty $\sigma(\cdot)$ is high), and its global maximiser is used as the next experimental setting.

Acquisition functions are designed to be large near *potentially* high values of the objective function. Figure 3 shows commonly used acquisition functions: PI, EI, and GP-UCB. PI prefers areas where improvement over the current maximum $f(\boldsymbol{x}^+)$ is most likely. EI considers not only probability of improvement, but also the expected magnitude of improvement. GP-UCB maximises $f(\cdot)$ while minimising regret, the difference between the average utility and the ideal utility. Regret bounds are important for theoretically proving convergence. Unlike the original function, the acquisition function can be cheaply sampled, and may be optimised using a derivative-free global optimisation method like DIRECT [34] or using multi-start method with a derivative based local optimiser such as L-BFGS [35]. Details can be found in [19], [21].

III. EXPERIMENTAL DESIGN WITH BAYESIAN OPTIMISATION

BO has been influential in computer science for hyperparameter tuning [38]–[42], combinatorial optimisation [43], [44], and reinforcement learning [21]. Recent years have seen new applications in areas such as robotics [45], [46], neuroscience [47], [48], and materials discovery [49]–[55].

Bayesian optimisation is an iterative process outlined in Figure 2, which can be applied to experiments where inputs are unconstrained and the objective is a scalarised function of measured outputs. Examples of this kind include material design using physical models [56], or laboratory experiments [25]. However, experiments often involve

Acquisition Function	Formulation	Reference
Probability of Improvement	$\mathrm{PI}(\boldsymbol{x}) = \Phi\left(\frac{\mu_t(\boldsymbol{x}) - f(\boldsymbol{x}_t^+) - \xi}{\sigma(x)}\right)$	[31]
Expected Improvement	$\mathrm{EI}(\mathbf{x}) = (\mu_t(\boldsymbol{x}) - f(\boldsymbol{x}_t^+))\Phi(Z) + \sigma_t(\boldsymbol{x})\phi(Z)$ where $Z = \frac{\mu_t(\boldsymbol{x}) - f(\boldsymbol{x}_t^+)}{\sigma_t(\boldsymbol{x})}$	[32]
GP Upper Confidence Bound	$\text{GP-UCB}(\boldsymbol{x}) = \mu_t(\boldsymbol{x}) + \kappa_t \sigma_t(\boldsymbol{x})$	[33]

FIGURE 3. Acquisition functions expressed in terms of the mean $\mu(x)$, variance $\sigma(x)$, and current maximum $f(x^+)$. $\Phi(\cdot)$ and $\phi(\cdot)$ are the cumulative distribution function and the probability distribution function of the standard normal distribution. Some functions include factors to balance between exploration and exploitation: ξ in PI is constant, whereas κ_t in GP-UCB usually increases with iteration, causing the search to maintain exploration even with many samples.

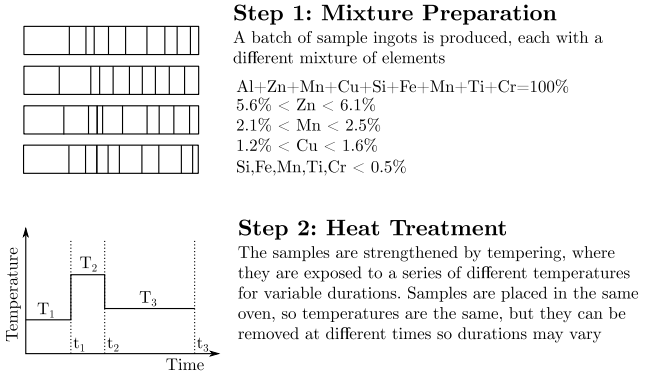


FIGURE 4. Case Study: Alloy design process. Alloy samples are cast with varying compositions according to a set of constraints. The samples are then tempered to improve hardness, which is subsequently measured. The physics of tempering of an alloy is based on nucleation and growth. During nucleation, new "phases" or precipitates are formed when clusters of atoms self-organise. These precipitates then diffuse together to achieve the requisite alloy characteristics in the growth step.

complicating factors such as constraints, batches, and multiple objectives. For example, in the alloy design process the composition of each sample follows a set of mixture constraints (see Figure 4). Batches of samples then undergo heat treatment for up to 70 hours, exposed to the same temperatures but with possible variation in duration between samples [24]. The optimiser must produce a batch of experimental settings, obeying inequality constraints, with some factors varying and others fixed within each batch. This impacts the design of the optimiser, through the formulation of the model, acquisition functions, and the search strategy. These are active areas of research, and recent developments are surveyed in the following discussion.

A. INCORPORATING PRIOR KNOWLEDGE

Where successive experiments are sufficiently similar to previous ones, it may be desirable to transfer knowledge from previous outcomes. Prior knowledge about the function or data can be used to reduce the search complexity and accelerate optimisation. Table 1 outlines some approaches. (1) Knowledge may be transferred from past (source) experiments to new (target) experiments where there are known or learnable similarities between the domains. For example, the source and target may be loosely similar, or have similar trends. (2) Where something is known about the influence of particular variables on the objective function, this can be imposed on the GP model. This could include monotonicity, function shape, or the probable location of the optimum or other features. (3) Where dependency structures exist in the design space, these can exploited to constrain the GP, or to handle high dimensionality via embedding.

B. HIGH DIMENSIONAL OPTIMISATION

The acquisition function must be optimised to find the next best suggestion for evaluating the objective. In continuous domain the acquisition functions can be extremely sharp in high dimensions, having only a few peaks marooned in a large terrain of almost flat surface. Global optimisation algorithms such as DIRECT [34] are infeasible above about 10 dimensions, and gradient-dependent methods cannot move if initialised in the flat terrain.

General strategies for tackling high-dimensionality include [103]: reducing the design space, screening important variables, decomposing the design into simpler sub-problems, mapping into a lower-dimensional space, and visualisation. Table 1(4) outlines approaches that have been reported for high dimensional BO, including: using coarse-to-fine approximations, projection into a lower-dimensional space, and approximation through low-rank matrices or additive structures. Choice of a method depends on whether the objective function has an intrinsic low dimensional structure (4B) or not (4A).

Standard BO is known to perform well in low dimensions, but performance degrades above about 15-20 dimensions. High dimensional BO has been demonstrated for 25-34 intrinsic dimensions on "real world" data, and up to 50 dimensions for synthetic functions [73], [77]. Projection methods have been shown to work independently of the number of extrinsic dimensions [43], [79], [81], whereas special kernels are shown to work in hundreds of dimensions [75].

C. MULTI-OBJECTIVE OPTIMISATION

Design problems often include multiple objectives which can be challenging to optimise. For example [104] demonstrates multiple objectives for discovery of new materials. Scalarisation by weighted sum of objectives can be done, but may not work when objectives have strong conflicts. In that setting a Pareto set of optimal points can be found [105]. For a point in a Pareto set, any one of the objectives cannot be improved without penalising another objective.

TABLE 1. Methods for transferring prior knowledge from past experiments (source) to new experiments (target) (1–3). Methods marked (*) have only been demonstrated for Gaussian processes, but are also applicable to Bayesian optimisation. Methods for handling high dimensionality (4), constraints (5), and parallel optimisation (6).

1. Transfer Learning (Overall shape using source functions or data)	
A) Source & target functions are loosely similar	(a) Merge source & target data after standardisation [57]
	(b) Merge source data as noisy observations into target [58, 59]
B) Difference between source & target can be learned as a simple function	Adjust source data through modelled difference function and then merge [59]
C) Source & target have similar overall trends	Source & target data are transformed and merged in latent space employing common rank [60]
D) Multiple similar sources, and similarity in target can be estimated	Similarity between source & target is incorporated in the kernel. GP jointly models all functions. Suitable for concurrent optimisation [41]
2. Transfer Learning (Trend per Variable)	
A) Monotonicity or convexity in any variable	(monotonic) 1st derivative is imposed to be positive or negative in GP [61, 62, 63]
	(convex/concave) 2nd order derivative is imposed to be positive or negative in GP [62, 64]
B) U or S shape in any variable	* Impose suitable constraints on first / second derivative. [64]
C) Unimodal in any variable	Estimating inflection point using a probabilistic model over inflection point [65]
D) Target optimum and source optimum are close	A prior belief on the optimum location is either estimated from source data or can be supplied by the user [66]
3. Transfer Learning (Dependency structure in design space)	
A) Known tree structure dependency in input space	a) Use tree structure to transfer information across overlapping tree paths [67]
	b) Use cylindrical embedding of input space [68]
B) Multiple sets of inputs independent of each other	a) *Additive structure imposed on kernel [69]
	b) Use method (a) for BO with known structure [70]
C) Input space has hierarchical structures	* Structure embedded through kernel design [71]
D) Structure of a process is known	The underlying function is composite [72]
4. High Dimensional Optimisation	
A) Function has no intrinsic low dimensional structure	a) Optimise acquisition function efficiently through "elastic" GP [73]
	b) Select random subset of variables in any iteration [74].
	c) Use cylindrical transformation of the high dimensional space [75]. However, it assumes that optimum is unlikely at the boundary.
B) Function has intrinsic low dimensional structure	a) Decompose into sub-spaces with smaller sub-sets. Prior is known: See 3B (Transfer Learning with "additive structure") Prior is unknown: Find latent space where model is decomposable [76, 77] Use method 3B(a) for BO with additive structure [78]
	b) Reduce dimensions by random projection into latent space [43, 79, 80, 81]
	c) Reduce dimensions through low rank matrix approximation [82]
5. Constraints	
A) Inequality Constraints	a) Weighted EI [83, 84], ADMMBO [85]
	b) Predictive Entropy Search [86, 87]
	c) Lookahead strategy [88]
B) Equality Constraints	d) Slack variable augmented Lagrangian [89]
C) Preference Constraints	e) Multiobjective optimisation with preference constraints [90]
6. Parallel (Batch) Optimisation	
A) No constraints within batch	a) Batch created through multi-step lookahead [91, 92, 93]
	b) Using heuristics for batch selection [94, 95, 96]
	c) Using multiple strategies Use of GP-UCB (1st point) + PE (additional points) [97] Array of GP-UCB strategies [98] Array of different Gaussian process models [99]
	d) Information theoretic approach [100]
	e) Knowledge gradient [101]
B) Constraints within batch	Constraints are imposed on selected variables [102]

Many methods have been proposed for using Bayesian optimisation for multi-objective optimisation [106]–[109], but these suffer from computational limitations because the acquisition function generally requires computation for all objective functions and as the number of objective functions grow the computational cost grows exponentially.

Moving away from EI, the method of [109] allows the optimisation of multiple objectives without rank modelling for conflicting objectives, while also remaining scale-invariant toward different objectives. The method performs better than [107], but suffers in high dimensions and can be computationally expensive. Predictive entropy search is used by [110], allowing the different objectives to be decoupled, computing acquisition for subsets of objectives when required. The computational cost increases linearly with the number of objectives. The method of [111] can be used for

single- or multiple-objective optimisation, including in multiple inequality constraints and has been shown to be robust in highly constrained settings where the feasible design space is small.

D. CONSTRAINTS

Table 1(5) outlines some approaches to handling constraints. If constraints are known, they can be handled during optimisation of the acquisition function by limiting the search. More difficult are ''black box'' constraints that can be evaluated but have unknown form. If the constraint is cheap to evaluate, this is not a problem. Methods for expensive constraint functions include a weighted EI function [83], [84], and weighted predictive entropy search [86]. A lookahead strategy for unknown constraints is described by [88]. A different formulation for the unknown is proposed by [85], handling expensive constraints using ADMM solver of [112].

The above methods deal with inequality constraints. In [89] both inequality and equality constraints are handled, using slack variables to convert inequality constraints to equality constraints, and Augmented Lagrangian (AL) to convert these inequality constraints into a sequence of simpler sub-problems.

The concept of weighted predictive entropy search has been extended for multi-objective problems [87] for inequality constraints which are both unknown and expensive to evaluate. A different type of constraint specifically for multiple objectives is investigated by [90] where between all the objectives, there exists a rank order preference on which objective is important. The algorithm developed therein can preferentially sample the Pareto set such that Pareto samples are more varied for the more important objectives.

E. PARALLEL (BATCH) OPTIMISATION

In some experiments it can be efficient to evaluate several settings in parallel. For example, during alloy design batches of different mixtures undergo similar heat treatment phases, so the optimiser must recommend multiple settings before receiving any new results. Sequential algorithms can be used to find the point that maximises the acquisition function, and then move on to find the next point in the batch after suppressing this point. Suppression can be achieved by temporarily updating the GP with a hypothetical value for the point (e.g. based on a recent posterior mean), or by applying a penalty in the acquisition function. Table 1(6) outlines some approaches that have been reported. Most methods are for unconstrained batches, though recent work has handled constraints on selected variables within a batch [102].

F. MULTI-FIDELITY OPTIMISATION

When function evaluations are prohibitively expensive, cheap approximations may be useful. In such situations high fidelity data obtained through experimentation might be augmented by low fidelity data obtained through running a simulation. For example, during alloy design, simulation software can predict the alloy strength but results may be less accurate than measurements obtained from casting experiments. Multi-fidelity Bayesian optimisation has been demonstrated in [113], [114]. Recently, [115] proposed BO for an optimisation problem with multi-fidelity data. Although multi-fidelity approach has been applied in problem-specific context or non-optimisation related tasks [41], [116]–[120], the method of [115] generalises well for BO problems.

G. MIXED-TYPE INPUT

Experimental parameters are often combinations of different types: continuous, discrete, categorical, and binary. Incorporation of mixed type input is challenging across the domains, including simpler methods such as Latin hypercube sampling [11]. Non-continuous variables are problematic in BO because the objective function approximation with GP assumes continuous input space, with covariance functions defining the relationship between these continuous variables. One common way to deal with discrete variables is to round the value to a close integer [40], but this approach leads to sub-optimal optimisation [121].

Two options for handling mixed-type inputs are: (1) designing kernels that are suitable for different variables, and (2) subsampling of data for maximising the objective function, which is especially useful in higher dimensional space. For integer variables the problem can be solved through kernel transformation, by assuming the objective function to be flat for the region where two continuous variables would be rounded to the same integer [121]. In [67] categorical variables are included by one-hot-encoding alongside numerical variables. A specialised kernel for categorical variables is proposed in [122].

Random forest regression is a good alternative to GP for regression in a sequential model-based algorithm configuration (SMAC, [44]). Random forests are good at exploitation but don't perform well for exploration as they may not predict well at points that are distant from observations. Additionally, a non-differentiable response surface renders it unsuitable for gradient-based optimisation.

IV. DISCUSSION

Machine-learning methods through Bayesian optimisation offer a powerful way to deal with many problems of experimental optimization that have not been previously addressed. While techniques exist for different issues (high dimensionality, multi-objective, etc.), few works solve multiple issues in a general way. Methods are likely to be composable where no incompatible changes are required to the BO process. Figure 5 outlines composability based on the current repertoire of Bayesian optimisation algorithms. When a design problem is single objective, has single fidelity measurement, and all the variables are continuous then it offers the greatest flexibility in terms of adding specific capability such as transfer learning or high dimensional optimisation. Other cases require careful selection of algorithms to add desired capabilities. For example, the method of [111] handles multiple objectives with constraints, and the method

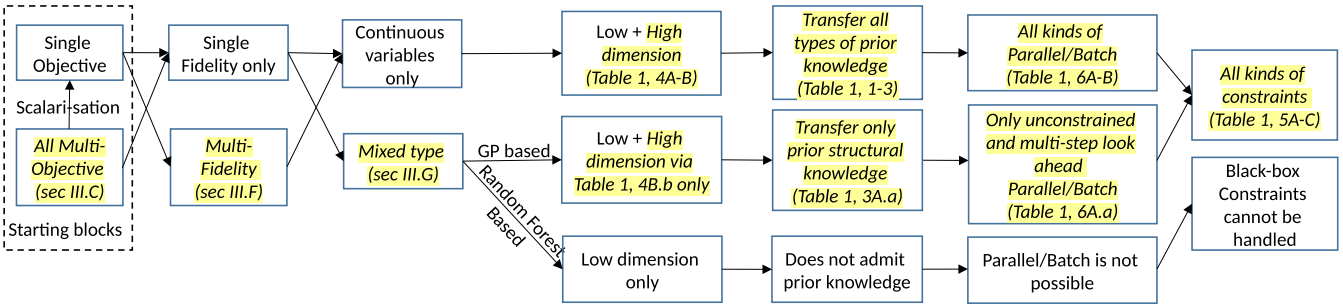


FIGURE 5. Current capability graph on the composability of various aspects of experimental design problems in Bayesian optimisation. It is possible to compose algorithms which lie on a path in the graph. It is possible to finish at any block and even skip multiple blocks on a path. Regular text denotes the capability achievable with standard Bayesian optimisation, whereas highlighted text denotes the existence of specialised algorithms.

of [43] handles parallel evaluation in high dimensions with mixed type inputs. Some combinations may not even be possible, for example, Random Forest based algorithm such as [44] would not admit many capabilities. Note that this graph does not portray any theoretical limitations, but merely presents a gist of the current capability through the lens of composability.

Several open-source libraries are available for incorporating BO into computer programs. Depending on the application, computation speed may be an issue. A common operation in most algorithms is Cholesky decomposition which is used to invert the kernel matrix and is generally $O(n^3)$ for n data points, but with care this can be calculated incrementally as new points arrive, reducing the complexity to $O(n^2)$ [123]. Several algorithms gain speed-up by implementing part of the algorithm on a GPU, which can be up to 100 times faster than the equivalent single-threaded code [124].

- GPyOpt (https://github.com/SheffieldML/GPyOpt) is a Bayesian optimisation framework, written in Python and supporting parallel optimisation, mixed factor types (continuous, discrete, and categorical), and inequality constraints.
- GPflowOpt (https://github.com/GPflow/GPflowOpt) is written in Python and uses TensorFlow (https://www.tensorflow.org) to accelerate computation on GPU hardware. It supports multi-objective acquisition functions, and black-box constraints [125].
- DiceOptim (https://cran.r-project.org/web/packages/DiceOptim/index.html) is a BO package written in R. Mixed equality and inequality constraints are implemented using the method of [89], and parallel optimisation is via multipoint EI [91], however parallel and constraints cannot be mixed in a single optimisation.
- MOE (https://github.com/Yelp/MOE) supports parallel optimisation via multi-point stochastic gradient ascent [124]. Interfaces are provided for Python and C++, and optimisation can be accelerated on GPU hardware.
- SigOpt (http://sigopt.com) offers Bayesian optimisation as a web service. The implementation is based on MOE, but includes some enhancements such as mixed factor types (continuous, discrete, categorical), and automatic hyperparameter tuning.
- BayesOpt (https://github.com/rmcantin/bayesopt) is written in C++, and includes common interfaces for C, C++, Python, Matlab, and Octave [123].

V. CONCLUSION

This review has presented an overview of Bayesian optimisation (BO) with application to experimental design. BO was introduced in relation to existing Design of Experiments (DOE) methods such as factorial designs, response surface methodology, and adaptive sampling. A brief discussion of the theory highlighted the roles of the Gaussian process, kernel, and acquisition function. A set of seven core issues was identified as being important in practical experimental designs, and some detailed solutions were reviewed. These core issues are: (1) the incorporation of prior knowledge, (2) high dimensional optimisation, (3) constraints, (4) batch evaluation, (5) multiple objectives, (6) multi-fidelity data, and (7) mixed variable types.

Recent works have shown the potential of Bayesian optimisation in fields such as robotics, neuroscience, and materials discovery. As the range of potential applications expands, it is increasingly unlikely that ''vanilla'' optimisation approaches for small numbers of unconstrained, continuous variables will be appropriate. This is particularly true in DACE simulation applications where high dimensional mixed-type inputs are typical.

Bayesian optimisation offers a powerful and rigorous framework for exploring and optimising expensive ''black box'' functions. While solutions exist for the core issues in experimental design, each approach has strengths and weaknesses that could potentially be improved, and the combination of the individual solutions is not necessarily straightforward. Thus there is a need for ongoing work in this area to: (1) improve the efficiency, generality, and scalability of approaches to the core issues, (2) develop designs that allow easy combination of multiple approaches, and (3) develop theoretical guarantees on the performance of solutions.

REFERENCES

[1] R. A. Fisher, *The Design of Experiments*. Edinburgh, U.K.: Oliver & Boyd, 1935.

[2] D. C. Montgomery, *Design and Analysis of Experiments*. Hoboken, NJ, USA: Wiley, 2017.

[3] R. H. Myers, D. C. Montgomery, and C. M. Anderson-Cook, "Response surface methodology: Process and product optimization using designed experiments," in *Applied Probability & Statistics* (Wiley Series in Probability and Statistics). 2009.

[4] G. G. Wang and S. Shan, "Review of metamodeling techniques in support of engineering design optimization," *J. Mech. Des.*, vol. 129, no. 4, pp. 370–380, Apr. 2007.

[5] W. C. Parr, "Introduction to quality engineering: Designing quality into products and processes," *Technometrics*, vol. 31, no. 2, pp. 255–256, May 1989.

[6] T. Hasenkamp, M. Arvidsson, and I. Gremyr, "A review of practices for robust design methodology," *J. Eng. Design*, vol. 20, no. 6, pp. 645–657, Dec. 2009.

[7] S. M. Göhler and T. J. Howard, "A framework for the application of robust design methods and tools," in *Proc. 1st Int. Symp. Robust Design*, T. J. Howard and T. Eifler, Eds. Lyngby, Denmark: Technical Univ. of Denmark, 2014, pp. 123–133.

[8] D. C. Woods and S. M. Lewis, "Design of experiments for screening," in *Handbook Uncertainty Quantification*. 2017, pp. 1143–1185.

[9] A. Dean and S. Lewis, Eds., *Screening: Methods for Experimentation in Industry, Drug Discovery, and Genetics*. New York, NY, USA: Springer-Verlag, 2006.

[10] F. A. Viana, "Things you wanted to know about the Latin hypercube design and were afraid to ask," in *Proc. 10th World Congr. Struct. Multidisciplinary Optim.*, 2013, pp. 1–9.

[11] H. Vieira, S. M. Sanchez, K. H. Kienitz, and M. C. N. Belderrain, "Efficient, nearly orthogonal-and-balanced, mixed designs: An effective way to conduct trade-off analyses via simulation," *J. Simul.*, vol. 7, no. 4, pp. 264–275, Nov. 2013.

[12] G. E. P. Box and K. B. Wilson, "On the experimental attainment of optimum conditions," *J. Roy. Stat. Soc. B, Methodol.*, vol. 13, no. 1, pp. 1–45, 1951.

[13] J. L. Chapman, L. Lu, and C. M. Anderson-Cook, "Process optimization for multiple responses utilizing the Pareto front approach," *Qual. Eng.*, vol. 26, no. 3, pp. 253–268, Jul. 2014.

[14] R. H. Myers, D. C. Montgomery, G. G. Vining, C. M. Borror, and S. M. Kowalski, "Response surface methodology: A retrospective and literature survey," *J. Qual. Technol.*, vol. 36, no. 1, pp. 53–77, Jan. 2004.

[15] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, "Design and analysis of computer experiments," *Stat. Sci.*, vol. 4, no. 4, pp. 409–423, 1989.

[16] H. Liu, Y.-S. Ong, and J. Cai, "A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design," *Struct. Multidisciplinary Optim.*, vol. 57, no. 1, pp. 393–416, Jan. 2018.

[17] F. A. Viana, T. W. Simpson, V. Balabanov, and V. Toropov, "Special section on multidisciplinary design optimization: Metamodeling in multidisciplinary design optimization: How far have we really come?" *AIAA J.*, vol. 52, no. 4, pp. 670–690, 2014.

[18] S. S. Garud, I. A. Karimi, and M. Kraft, "Design of computer experiments: A review," *Comput. Chem. Eng.*, vol. 106, pp. 71–95, Nov. 2017.

[19] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Global Optim.*, vol. 13, no. 4, pp. 455–492, 1998.

[20] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of Bayesian optimization," *Proc. IEEE*, vol. 104, no. 1, pp. 148–175, Jan. 2016.

[21] E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," 2010, *arXiv:1012.2599*. [Online]. Available: https://arxiv.org/abs/1012.2599

[22] P. I. Frazier, "A tutorial on Bayesian optimization," 2018, *arXiv:1807.02811*. [Online]. Available: https://arxiv.org/abs/1807.02811

[23] D. Xue, P. V. Balachandran, J. Hogden, J. Theiler, D. Xue, and T. Lookman, "Accelerated search for materials with targeted properties by adaptive design," *Nature Commun.*, vol. 7, p. 11241, Apr. 2016.

[24] A. Vahid, S. Rana, S. Gupta, P. Vellanki, S. Venkatesh, and T. Dorin, "New Bayesian-optimization-based design of high-strength 7xxx-series alloys from recycled aluminum," *J. Minerals, Metals Mater. Soc.*, vol. 70, no. 11, pp. 2704–2709, Nov. 2018.

[25] C. Li, D. R. de Celis Leal, S. Rana, S. Gupta, A. Sutti, S. Greenhill, T. Slezak, M. Height, and S. Venkatesh, "Rapid Bayesian optimisation for synthesis of short polymer fiber materials," *Sci. Rep.*, vol. 7, no. 1, p. 5683, 2017.

[26] S. Ju, T. Shiga, L. Feng, Z. Hou, K. Tsuda, and J. Shiomi, "Designing nanostructures for phonon transport via Bayesian optimization," *Phys. Rev. X*, vol. 7, no. 2, 2017, Art. no. 021024.

[27] H. Abdelrahman, F. Berkenkamp, J. Poland, and A. Krause, "Bayesian optimization for maximum power point tracking in photovoltaic power plants," in *Proc. Eur. Control Conf. (ECC)*, Jun. 2016, pp. 2078–2083.

[28] S. Kikuchi, H. Oda, S. Kiyohara, and T. Mizoguchi, "Bayesian optimization for efficient determination of metal oxide grain boundary structures," *Phys. B, Condens. Matter*, vol. 532, pp. 24–28, Mar. 2018.

[29] M. M. Khajah, B. D. Roads, R. V. Lindsey, Y.-E. Liu, and M. C. Mozer, "Designing engaging games using Bayesian optimization," in *Proc. CHI Conf. Hum. Factors Comput. Syst. (CHI)*, 2016, pp. 5571–5582.

[30] R. Lorenz, I. R. Violante, R. P. Monti, G. Montana, A. Hampshire, and R. Leech, "Dissociating frontoparietal brain networks with neuroadaptive Bayesian optimization," *Nature Commun.*, vol. 9, no. 1, p. 1227, 2018.

[31] H. J. Kushner, "A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise," *J. Basic Eng.*, vol. 86, no. 1, pp. 97–106, Mar. 1964.

[32] J. Mockus, V Tiesis, and A. Zilinskas, "The application of Bayesian methods for seeking the extremum," in *Toward Global Optimization 2*, L. C. W. Dixon and G. P Szego, Eds. Amsterdam, The Netherlands: North Holland, 1978.

[33] N. Srinivas, A. Krause, S. Kakade, and M. W. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 1015–1022.

[34] D. R. Jones, C. D. Perttunen, and B. E. Stuckman, "Lipschitzian optimization without the Lipschitz constant," *J. Optim. Theory Appl.*, vol. 79, no. 1, pp. 157–181, 1993.

[35] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Math. Program.*, vol. 45, nos. 1–3, pp. 503–528, 1989.

[36] M. J. Powell, "A view of algorithms for optimization without derivatives," *Math. Today-Bull. Inst. Math. Appl.*, vol. 43, no. 5, pp. 170–174, 2007.

[37] T. P. Runarsson and X. Yao, "Stochastic ranking for constrained evolutionary optimization," *IEEE Trans. Evol. Comput.*, vol. 4, no. 3, pp. 284–294, Sep. 2000.

[38] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2546–2554.

[39] C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 2013, pp. 847–855.

[40] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2960–2968.

[41] K. Swersky, J. Snoek, and R. P. Adams, "Multi-task Bayesian optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2004–2012.

[42] P. Baldi, P. Sadowski, and D. Whiteson, "Enhanced Higgs boson to $\tau+\tau$-search with deep learning," *Phys. Rev. Lett.*, vol. 114, no. 11, 2015, Art. no. 111801.

[43] Z. Wang, M. Zoghi, F. Hutter, D. Matheson, and N. De Freitas, "Bayesian optimization in high dimensions via random embeddings," in *Proc. Int. Joint Conf. Artif. Intell.*, Jul. 2013, pp. 1778–1784.

[44] F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Sequential model-based optimization for general algorithm configuration," in *Proc. Int. Conf. Learn. Intell. Optim.* Berlin, Germany: Springer, 2011, pp. 507–523.

[45] A. Cully, J. Clune, D. Tarapore, and J.-B. Mouret, "Robots that can adapt like animals," *Nature*, vol. 521, no. 7553, pp. 503–507, May 2015.

[46] R. Calandra, A. Seyfarth, J. Peters, and M. P. Deisenroth, "Bayesian optimization for learning gaits under uncertainty," *Ann. Math. Artif. Intell.*, vol. 76, nos. 1–2, pp. 5–23, Feb. 2016.

[47] J. Lancaster, R. Lorenz, R. Leech, and J. H. Cole, "Bayesian optimisation for neuroimaging pre-processing in brain age classification and prediction," *Frontiers Aging Neurosci.*, vol. 10, p. 28, Feb. 2018.

[48] R. Lorenz, R. P. Monti, I. R. Violante, C. Anagnostopoulos, A. A. Faisal, G. Montana, and R. Leech, "The automatic neuroscientist: A framework for optimizing experimental design with closed-loop real-time fMRI," *NeuroImage*, vol. 129, pp. 320–334, Apr. 2016.

[49] T. Ueno, T. D. Rhone, Z. Hou, T. Mizoguchi, and K. Tsuda, "COMBO: An efficient Bayesian optimization library for materials science," *Mater. Discovery*, vol. 4, pp. 18–21, Jun. 2016.
[50] T. Lookman, P. V. Balachandran, D. Xue, J. Hogden, and J. Theiler, "Statistical inference and adaptive design for materials discovery," *Current Opinion Solid State Mater. Sci.*, vol. 21, no. 3, pp. 121–128, Jun. 2017.
[51] R. Gómez-Bombarelli *et al.*, "Design of efficient molecular organic light-emitting diodes by a high-throughput virtual screening and experimental approach," *Nature Mater.*, vol. 15, no. 10, pp. 1120–1127, Oct. 2016.
[52] P. I. Frazier and J. Wang, "Bayesian optimization for materials design," in *Proc. Inf. Sci. Mater. Discovery Design*. Cham, Switzerland: Springer, 2016, pp. 45–75.
[53] A. Seko, T. Maekawa, K. Tsuda, and I. Tanaka, "Machine learning with systematic density-functional theory calculations: Application to melting temperatures of single-and binary-component solids," *Phys. Rev. B, Condens. Matter*, vol. 89, no. 5, 2014, Art. no. 054303.
[54] A. Seko, A. Togo, H. Hayashi, K. Tsuda, L. Chaput, and I. Tanaka, "Discovery of low thermal conductivity compounds with first-principles anharmonic lattice dynamics calculations and Bayesian optimization," 2015, *arXiv:1506.06439*. [Online]. Available: https://arxiv.org/abs/1506.06439
[55] A. Seko, A. Togo, H. Hayashi, K. Tsuda, L. Chaput, and I. Tanaka, "Prediction of low-thermal-conductivity compounds with first-principles anharmonic lattice-dynamics calculations and Bayesian optimization," *Phys. Rev. Lett.*, vol. 115, no. 20, 2015, Art. no. 205901.
[56] D. Packwood, *Bayesian Optimization for Materials Science*. Singapore: Springer, 2017.
[57] D. Yogatama and G. Mann, "Efficient transfer learning method for automatic hyperparameter tuning," in *Proc. 17th Int. Conf. Artif. Intell. Statist. (AISTATS)*, Reykjavik, Iceland, Apr. 2014, pp. 1077–1085.
[58] T. T. Joy, S. Rana, S. K. Gupta, and S. Venkatesh, "Flexible transfer learning framework for Bayesian optimisation," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*. Cham, Switzerland: Springer, 2016, pp. 102–114.
[59] A. Shilton, S. Gupta, S. Rana, and S. Venkatesh, "Regret bounds for transfer learning in Bayesian optimisation," in *Proc. Artif. Intell. Statist.*, 2017, pp. 307–315.
[60] R. Bardenet, M. Brendel, B. Kégl, and M. Sebag, "Collaborative hyperparameter tuning," in *Proc. 30th Int. Conf. Mach. Learn. (ICML)*, Atlanta, GA, USA, Jun. 2013, pp. 199–207.
[61] J. Riihimäki and A. Vehtari, "Gaussian processes with monotonicity information," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 645–652.
[62] M. Jauch and V. Peña, "Bayesian optimization with shape constraints," 2016, *arXiv:1612.08915*. [Online]. Available: https://arxiv.org/abs/1612.08915
[63] C. Li, S. Rana, S. Gupta, V. Nguyen, and S. Venkatesh, "Bayesian optimization with monotonicity information," in *Proc. 31st Conf. Neural Inf. Process. Syst. (NIPS)*, 2017.
[64] P. J. Lenk and T. Choi, "Bayesian analysis of shape-restricted functions using Gaussian process priors," *Statistica Sinica*, vol. 27, pp. 43–69, Jan. 2017.
[65] M. R. Andersen, E. Siivola, and A. Vehtari, "Bayesian optimization of unimodal functions," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2017.
[66] A. Ramachandran, S. K. Gupta, R. Santu, and S. Venkatesh, "Information-theoretic transfer learning framework for Bayesian optimisation," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Cham, Switzerland: Springer, 2018.
[67] R. Jenatton, C. Archambeau, J. González, and M. Seeger, "Bayesian optimization with tree-structured dependencies," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1655–1664.
[68] K. Swersky, D. Duvenaud, J. Snoek, F. Hutter, and M. A. Osborne, "Raiders of the lost architecture: Kernels for Bayesian optimization in conditional parameter spaces," 2014, *arXiv:1409.4011*. [Online]. Available: https://arxiv.org/abs/1409.4011
[69] D. K. Duvenaud, H. Nickisch, and C. E. Rasmussen, "Additive Gaussian processes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 226–234.
[70] K. Kandasamy, J. G. Schneider, and B. Póczos, "High dimensional Bayesian optimisation and bandits via additive models," in *Proc. 32nd Int. Conf. Mach. Learn. (ICML)*, Lille, France, Jul. 2015, pp. 295–304.
[71] F. Hutter and M. A. Osborne, "A kernel for hierarchical parameter spaces," 2013, *arXiv:1310.5738*. [Online]. Available: https://arxiv.org/abs/1310.5738
[72] T. Dai Nguyen, S. Gupta, S. Rana, V. Nguyen, S. Venkatesh, K. J. Deane, and P. G. Sanders, *Cascade Bayesian Optimization*. 2016, pp. 268–280.
[73] S. Rana, C. Li, S. Gupta, V. Nguyen, and S. Venkatesh, "High dimensional Bayesian optimization with elastic Gaussian process," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2883–2891.
[74] C. Li, S. Gupta, S. Rana, V. Nguyen, S. Venkatesh, and A. Shilton, "High dimensional Bayesian optimization using dropout," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, 2017, pp. 2096–2102.
[75] C. Oh, E. Gavves, and M. Welling, "BOCK: Bayesian optimization with cylindrical kernels," in *Proc. 35th Int. Conf. Mach. Learn.*, J. Dy and A. Krause, Eds. Stockholm, Sweden: Stockholmsmässan, 2018, pp. 3868–3877.
[76] C.-L. Li, K. Kandasamy, B. Póczos, and J. Schneider, "High dimensional Bayesian optimization via restricted projection pursuit models," in *Proc. Artif. Intell. Statist.*, 2016, pp. 884–892.
[77] Z. Wang, C. Li, S. Jegelka, and P. Kohli, "Batched high-dimensional Bayesian optimization via structural kernel learning," in *Proc. Int. Conf. Mach. Learn.*, 2017.
[78] J. Gardner, C. Guo, K. Weinberger, R. Garnett, and R. Grosse, "Discovering and exploiting additive structure for Bayesian optimization," in *Proc. Artif. Intell. Statist.*, 2017, pp. 1311–1319.
[79] A. Nayebi, A. Munteanu, and M. Poloczek, "A framework for Bayesian optimization in embedded subspaces," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 4752–4761.
[80] M. Mutny and A. Krause, "Efficient high dimensional Bayesian optimization with additivity and quadrature Fourier features," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 9005–9016.
[81] J. Kirschner, M. Mutny, N. Hiller, R. Ischebeck, and A. Krause, "Adaptive and safe Bayesian optimization in high dimensions via one-dimensional subspaces," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 3429–3438.
[82] J. Djolonga, A. Krause, and V. Cevher, "High-dimensional Gaussian process bandits," in *Proc. Adv. Neural Inf. Process. Syst. 27th Annu. Conf. Neural Inf. Process. Syst.*, Lake Tahoe, NV, USA, 2013, pp. 1025–1033.
[83] M. A. Gelbart, J. Snoek, and R. P. Adams, "Bayesian optimization with unknown constraints," in *Proc. Uncertainty Artif. Intell.*, 2014, pp. 250–259.
[84] J. R. Gardner, M. J. Kusner, Z. E. Xu, K. Q. Weinberger, and J. P. Cunningham, "Bayesian optimization with inequality constraints," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 937–945.
[85] S. Ariafar, J. Coll-Font, D. Brooks, and J. Dy, "ADMMBO: Bayesian optimization with unknown constraints using ADMM," *J. Mach. Learn. Res.*, vol. 20, no. 123, pp. 1–26, 2019.
[86] J. M. Hernández-Lobato, M. Gelbart, M. Hoffman, R. Adams, and Z. Ghahramani, "Predictive entropy search for Bayesian optimization with unknown constraints," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1699–1707.
[87] E. C. Garrido-Merchán and D. Hernández-Lobato, "Predictive entropy search for multi-objective Bayesian optimization with constraints," *Neurocomputing*, vol. 361, pp. 50–68, Oct. 2019.
[88] R. Lam and K. Willcox, "Lookahead Bayesian optimization with inequality constraints," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1888–1898.
[89] V. Picheny, R. B. Gramacy, S. Wild, and S. Le Digabel, "Bayesian optimization under mixed constraints with a slack-variable augmented Lagrangian," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1435–1443.
[90] M. Abdolshah, A. Shilton, S. Rana, S. Gupta, and S. Venkatesh, "Multi-objective Bayesian optimisation with preferences over objectives," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2019.
[91] D. Ginsbourger, R. Le Riche, and L. Carraro, "A multi-points criterion for deterministic parallel global optimization based on Gaussian processes," Département Méthodes et Modèles Mathématiques pour l'Industrie, 3MI-ENSMSE, Saint-Étienne, France, Tech. Rep. hal-00260579, 2008.
[92] J. Azimi, A. Fern, and X. Z. Fern, "Batch Bayesian optimization via simulation matching," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 109–117.
[93] T. Desautels, A. Krause, and J. W. Burdick, "Parallelizing exploration-exploitation tradeoffs in Gaussian process bandit optimization," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3873–3923, 2014.
[94] J. González, Z. Dai, P. Hennig, and N. D. Lawrence, "Batch Bayesian optimization via local penalization," in *Proc. Artif. Intell. Statist.*, 2015, pp. 648–657.
[95] V. Nguyen, S. Rana, S. K. Gupta, C. Li, and S. Venkatesh, "Budgeted batch Bayesian optimization," in *Proc. IEEE 16th Int. Conf. Data Mining (ICDM)*, Dec. 2016, pp. 1107–1112.

[96] C. Gong, J. Peng, and Q. Liu, "Quantile stein variational gradient descent for batch Bayesian optimization," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 2347–2356.
[97] E. Contal, D. Buffoni, A. Robicquet, and N. Vayatis, "Parallel Gaussian process optimization with upper confidence bound and pure exploration," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Berlin, Germany: Springer, 2013, pp. 225–240.
[98] S. Gupta, A. Shilton, S. Rana, and S. Venkatesh, "Exploiting strategy-space diversity for batch Bayesian optimization," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2018, pp. 538–547.
[99] T. T. Joy, S. Rana, S. Gupta, and S. Venkatesh, "Batch Bayesian optimization using multi-scale search," *Knowl.-Based Syst.*, vol. 187, Jan. 2020, Art. no. 104818.
[100] A. Shah and Z. Ghahramani, "Parallel predictive entropy search for batch global optimization of expensive objective functions," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 3330–3338.
[101] J. Wu and P. Frazier, "The parallel knowledge gradient method for batch Bayesian optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3126–3134.
[102] P. Vellanki, S. Rana, S. Gupta, D. Rubin, A. Sutti, T. Dorin, M. Height, P. Sanders, and S. Venkatesh, "Process-constrained batch Bayesian optimisation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3417–3426.
[103] S. Shan and G. G. Wang, "Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions," *Struct. Multidisciplinary Optim.*, vol. 41, no. 2, pp. 219–241, Mar. 2010.
[104] A. M. Gopakumar, P. V. Balachandran, D. Xue, J. E. Gubernatis, and T. Lookman, "Multi-objective optimization for materials discovery via adaptive design," *Sci. Rep.*, vol. 8, no. 1, p. 3738, 2018.
[105] Y. Collette and P. Siarry, *Multiobjective Optimization: Principles and Case Studies*. Berlin, Germany: Springer-Verlag, 2013.
[106] J. Knowles, "ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 10, no. 1, pp. 50–66, Feb. 2006.
[107] W. Ponweiser, T. Wagner, D. Biermann, and M. Vincze, "Multiobjective optimization on a limited budget of evaluations using model-assisted $\mathcal{S}$-metric selection," in *Proc. Int. Conf. Parallel Problem Solving Nature*. Berlin, Germany: Springer, 2008, pp. 784–794.
[108] M. Emmerich and J.-W. Klinkenberg, "The computation of the expected improvement in dominated hypervolume of Pareto front approximations," Rapport Technique, Leiden Univ., Leiden, The Netherlands, Tech. Rep. LIACS-TR 9-2008, 2008.
[109] V. Picheny, "Multiobjective optimization using Gaussian process emulators via stepwise uncertainty reduction," *Statist. Comput.*, vol. 25, no. 6, pp. 1265–1280, Nov. 2015.
[110] D. Hernández-Lobato, J. Hernandez-Lobato, A. Shah, and R. Adams, "Predictive entropy search for multi-objective Bayesian optimization," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1492–1501.
[111] P. Feliot, J. Bect, and E. Vazquez, "A Bayesian approach to constrained single-and multi-objective optimization," *J. Global Optim.*, vol. 67, nos. 1–2, pp. 97–133, 2017.
[112] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.
[113] P. Perdikaris and G. E. Karniadakis, "Model inversion via multi-fidelity Bayesian optimization: A new paradigm for parameter estimation in haemodynamics, and beyond," *J. Roy. Soc. Interface*, vol. 13, no. 118, p. 20151107, 2016.
[114] A. Marco, F. Berkenkamp, P. Hennig, A. P. Schoellig, A. Krause, S. Schaal, and S. Trimpe, "Virtual vs. real: Trading off simulations and physical experiments in reinforcement learning with Bayesian optimization," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 1557–1563.
[115] K. Kandasamy, G. Dasarathy, J. Schneider, and B. Poczos, "Multi-fidelity Bayesian optimisation with continuous approximations," in *Proc. Int. Conf. Mach. Learn.*, 2017.
[116] A. Klein, S. Bartels, S. Falkner, P. Hennig, and F. Hutter, "Towards efficient Bayesian optimization for big data," in *Proc. NIPS Workshop Bayesian Optim. (BayesOpt)*, vol. 134, 2015, p. 98.
[117] M. Poloczek, J. Wang, and P. Frazier, "Multi-information source optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4291–4301.
[118] M. Cutler, T. J. Walsh, and J. P. How, "Reinforcement learning with multi-fidelity simulators," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 3888–3895.
[119] A. Sabharwal, H. Samulowitz, and G. Tesauro, "Selecting near-optimal learners via incremental data allocation," in *Proc. AAAI*, 2016, pp. 2007–2015.
[120] C. Zhang and K. Chaudhuri, "Active learning from weak and strong labelers," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 703–711.
[121] E. C. Garrido-Merchán and D. Hernández-Lobato, "Dealing with Categorical and Integer-valued Variables in Bayesian optimization with Gaussian processes," 2017, *arXiv:1706.03673*. [Online]. Available: https://arxiv.org/abs/1805.03463
[122] M. A. Villegas García, "An investigation into new kernels for categorical variables," M.S. thesis, Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Barcelona, Spain, 2013.
[123] R. Martinez-Cantin, "BayesOpt: A Bayesian optimization library for nonlinear optimization, experimental design and bandits," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3735–3739, 2014.
[124] J. Wang, S. C. Clark, E. Liu, and P. I. Frazier, "Parallel Bayesian global optimization of expensive functions," 2016, *arXiv:1602.05149*. [Online]. Available: https://arxiv.org/abs/1602.05149
[125] N. Knudde, J. van der Herten, T. Dhaene, and I. Couckuyt, "GPflow: A Gaussian process library using TensorFlow," 2017, *arXiv:1711.03845*. [Online]. Available: https://arxiv.org/abs/1610.08733

STEWART GREENHILL received the B.Sc. degree in computer science from the University of Western Australia, in 1987, and the Ph.D. degree in environmental science from Murdoch University, in 1992. He is currently a Research Fellow with the Applied Artificial Intelligence Institute, Deakin University, Australia. His research interests include machine learning, signal processing, embedded systems, software engineering, visualization, and interaction design.

SANTU RANA is currently a Researcher in the field of machine learning and computer vision with the Applied Artificial Intelligence Institute, Deakin University, Australia. His research in high-dimensional Bayesian optimization has been applied to efficiently design alloys with large number of elements. He has been actively conducting research in Bayesian experimental design with applications in advanced manufacturing. In the last four years, he has published more than 40 research articles improving various aspects of Bayesian optimization algorithm. Altogether, he has published over 79 research articles, including 14 refereed journal articles, 58 fully refereed conference proceedings, and seven workshop articles, with over 515 citations and an H-index of 12. He is also a Co-Inventor of two patents. His broad research interests lie in devising practical machine learning algorithms for various tasks, such as object recognition, mathematical optimization, and healthcare data modeling.

SUNIL GUPTA is currently a Researcher in the field of machine learning and data mining with the Applied Artificial Intelligence Institute, Deakin University, Australia. He has published over 100 research articles, including two book chapters, 25 refereed journal articles, 70 fully refereed conference proceedings, and nine workshop articles with over 1000 citations and an H-index of 17. His research interest lies in developing data-driven models for real world processes and phenomena covering both big-data and small-data problems. His recent research in optimization using small data (Bayesian optimization) has found applications in efficient experimental design of products and processes in advanced manufacturing, such as alloy design with certain target properties, design of short nanofibers with appropriate length and thickness, and optimal setting of parameters in 3d-printers. His research has won several best paper awards in the field of data mining and machine learning. He is also a Co-Inventor of a patent related to experimental design. He regularly serves at technical program committees of the prestigious machine learning conferences.

PRATIBHA VELLANKI graduated from Pune University in electronics and telecommunication, in 2009, where she continued to pursue the master's degree in signal processing, in 2011. She received the Ph.D. degree from Deakin University, Australia. She worked as an Associate Research Fellow in the field of applied machine learning with Deakin University, from 2012 to 2018. She is interested in taking on a machine learning perspective on research problems that affect people and health. She currently works as a Data Scientist for the Office for National Statistics, U.K. This work was done during her time with Deakin University.

SVETHA VENKATESH is currently an ARC Australian Laureate Fellow, the Alfred Deakin Professor, and the Co-Director of the Applied Artificial Intelligence Institute (A2I2), Deakin University, Australia. She and her team have tackled a wide range of problems of societal significance, including the critical areas of autism, security, and aged care. The outcomes have impacted the community and evolved into publications, patents, tools, and spin-off companies. This includes more than 600 publications, three full patents, start-up companies (iCetana), and a significant product (TOBY Playpad).

Prof. Venkatesh was elected as a Fellow of the International Association of Pattern Recognition, in 2004, for contributions to formulation and extraction of semantics in multimedia data, and the Australian Academy of Technological Sciences and Engineering, in 2006. In 2017, she was appointed as an Australian Laureate Fellow, the highest individual award the Australian Research Council can bestow.