Preprint from dowlinglab.nd.edu, (c) 2021, Ke Wang and Alexander Dowling

Bayesian optimization for chemical products and functional materials

Ke Wang, Alexander W. Dowling*

Department of Chemical and Biomolecular Engineering
University of Notre Dame, Notre Dame, IN 46556

Abstract

The design of chemical-based products and functional materials is vital to modern technologies, yet remains expensive and slow. Artificial intelligence and machine learning offer new approaches to leverage data to overcome these challenges. This review focuses on recent applications of Bayesian optimization (BO) to chemical products and materials including molecular design, drug discovery, molecular modeling, electrolyte design, and additive manufacturing. Numerous examples show how BO often requires an order of magnitude fewer experiments than Edisonian search. The essential equations for BO are introduced in a self-contained primer specifically written for chemical engineers and others new to the area. Finally, the review discusses four current research directions for BO and their relevance to product and materials design.

Keywords: Bayesian optimization, machine learning, molecular design, product design, additive manufacturing, smart manufacturing

Why is materials and product design optimization so challenging?

Chemical-based products and materials with specific functionalities are ubiquitous in modern society and essential for many new technologies.[1, 2] Examples include coatings, fertilizers, food additives[3], medicines, engineered

*Corresponding author
Email address: adowling@nd.edu (Alexander W. Dowling)

gene sequences[4], and functional devices[5, 6]. Yet the design of chemical-based products and functional materials remains challenging. Often the design space is vast; for example, there are approximately 10^{23} drug-like molecules, of which only 10^8 have been synthesized.[7] Moreover, chemical products and functional materials cannot be designed in isolation; instead, multi-scale and multi-disciplinary design frameworks must account for technical, economic, social, and environmental factors across manufacturing, use, and end-of-life.[1] For example, engineering new functional materials (e.g., membranes) to enable enhanced separations (e.g., water treatment, CO_2 capture) should consider how the new material will be integrated into devices, systems, and infrastructures.[8] In computer-aided molecular design (CAMD), mathematical optimization is used to efficiently search through the vast molecular and materials design spaces to resolve (some) multiscale trade-offs.[9, 6] Historically, the success of CAMD has been limited by the predictive accuracy of physical property models. Recently developed machine/deep learning methods have shown great promise for predicting physical properties from data and are expected to revolutionize the computational design of chemical products and functional materials.[10]

What is Bayesian Optimization (BO)?

Bayesian optimization (BO) is a family of surrogate-assisted/derivative-free optimization algorithms that use Bayesian probability theory to explicitly balance trade-offs between exploitation and exploration.[11] BO has two core components: a computationally inexpensive stochastic surrogate model that emulates expensive computational or physical experiments and an acquisition function to determine the optimal sequence of future experiments. [12] BO is typically deployed in a feedback loop with experiments using the following general steps:

1. Identify the objective function(s) and decision variables (with bounds).
2. A space-filling design (e.g., Latin hypercube sample) is generated and experiments are performed.

3. The surrogate model is (re)trained using available experimental data.
4. Using the surrogate model, the acquisition function is maximized to recommend the next experiment(s).
5. Experiment(s) are performed and added to the training data.
6. If the goal has not been attained and the experimental budget has not been exhausted, GOTO step 3.

Gaussian Process Regression Surrogate Models

The goal of BO is to maximize $f(\cdot)$, an unknown function, using a probabilistic surrogate model.[13] Below we summarize Gaussian Processes (GP), which are the most popular surrogate models for BO, following the notation of [14]; also see [15] for an excellent introduction to GPs.

Let $\mathcal{D} = \{(\mathbf{x}_i, y_i), |\mathbf{x}_i \in \mathbb{R}^p, y_i \in \mathbb{R}, i \in 1, \ldots, n\}$ be a collection of n samples, where the vector $\mathbf{x}_i$ represents input variables (features) correspond to the observation y_i. For convenience, we denote the data $\mathcal{D} = (\mathbf{X}, \mathbf{y})$ using matrix $\mathbf{X} = (\mathbf{x_1}, \ldots, \mathbf{x_n}) \in \mathbb{R}^{n \times p}$ and vector $\mathbf{y} = (y_1, \ldots, y_n) \in \mathbb{R}^n$. We assume y_i includes measurement error that is normally distributed with zero mean and variance σ^2. Mathematically, $y_i = f(\mathbf{x}_i) + \varepsilon$ and $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ where $f(\cdot)$ is an unknown function.

A Gaussian Process can be thought of as a collection of normally distributed random variables that emulates the behavior of $f(\cdot) + \varepsilon$:

$$f \sim \mathcal{GP}(m(\mathbf{x} \mid \theta),\ k(\mathbf{x}, \mathbf{x}' \mid \theta)) \quad \mathbf{x}, \mathbf{x}' \in \mathbb{R}^p \tag{1}$$

Here $m(\mathbf{x} \mid \theta)$ is the mean function; although often set to zero for computer science applications (e.g., pattern recognition), the mean function is a natural way to incorporate physical relationships into the GP model. $k(\mathbf{x}, \mathbf{x}' \mid \theta)$ is the covariance or kernel function. θ are the hyperparameters, which are coefficients used to define the mean and kernel functions. Eq. (2) defines three popular kernel functions and their hyperparameters: Matérn (k_M), squared exponential (k_{SE}), and rational quadratic (k_{RQ}).[11] In Eq. (2a), $\nu = 5/2$ and $\nu = 3/2$ are the two most popular Matérn kernels and K_ν is a modified Bessel function.[16]

Often, $\ell \in \mathbb{R}^p$ is the length-scale for the features $\mathbf{x}$. Training hyperparameter ℓ gives insights into the most important dimensions of $\mathbf{x}$ and is known as automatic relevance determination. Alternatively, ℓ may be treated as a scalar, i.e., the same for each dimension of $\mathbf{x}$, to reduce the number of hyperparameters. Likewise, σ^2, the observation error variance, may be inferred from data as a hyperparameter. Alternatively, σ^2 may be fixed *a priori* by the modeler using knowledge of the experiments, e.g., quantified random error of the experimental apparatus.

$$k_{\mathrm{M}}\left(\mathbf{x}, \mathbf{x}' \mid \theta_M\right) = \frac{1}{\Gamma(\nu)2^{\nu-1}} \left[2\nu \sum_{j=1}^{p} \left(\frac{x_j - x_j'}{\ell_j}\right)^2\right]^{\nu/2} K_\nu \left[2\nu \sum_{j=1}^{p} \left(\frac{x_j - x_j'}{\ell_j}\right)^2\right]^{1/2},$$

$$\theta_M = (\ell) \tag{2a}$$

$$k_{SE}\left(\mathbf{x}, \mathbf{x}' \mid \theta_{SE}\right) = \exp\left(-\frac{1}{2} \sum_{j=1}^{p} \left(\frac{x_j - x_j'}{\ell_j}\right)^2\right), \quad \theta_{SE} = (\ell) \tag{2b}$$

$$k_{RQ}\left(\mathbf{x}, \mathbf{x}' \mid \theta_{RQ}\right) = \left[1 + \frac{1}{2\alpha} \sum_{j=1}^{p} \left(\frac{x_j - x_j'}{\ell_j}\right)^2\right]^{-\alpha}, \quad \theta_{RQ} = (\alpha, \ell) \tag{2c}$$

Given training data $(\mathbf{X}, \mathbf{y})$ and values of the hyperparameters θ and optionally the assumed observation error σ^2, one desires to predict $\mathbf{f}_*$ at new input values $\mathbf{X}_*$. The key mathematical property of a GP is that the outputs $\mathbf{y}$ and $\mathbf{f}_*$ follow a multivariate normal (Gaussian) distribution:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{m}(\mathbf{X} \mid \theta) \\ \mathbf{m}(\mathbf{X}_* \mid \theta) \end{bmatrix}, \begin{bmatrix} \mathbf{K}(\mathbf{X}, \mathbf{X} \mid \theta) + \sigma^2 \mathbf{I} & \mathbf{K}\left(\mathbf{X}, \mathbf{X}_* \mid \theta\right) \\ \mathbf{K}\left(\mathbf{X}_*, \mathbf{X} \mid \theta\right) & \mathbf{K}\left(\mathbf{X}_*, \mathbf{X}_* \mid \theta\right) \end{bmatrix}\right) \tag{3}$$

Here $\mathbf{K}(\cdot, \cdot \mid \theta)$ denotes the kernel matrix, which is the kernel function k evaluated elementwise.[14] Exploiting analytical properties of the multivariate normal distribution (see [14, 15] for details) gives the following expected value (mean) and (co)variance for the prediction $\mathbf{f}_*$ corresponding to the new input matrix $\mathbf{X}_*$:

$$\mu_*(\mathbf{X}_* \mid \theta) = \mathbb{E}(\mathbf{f}_*)$$
$$= \mathbf{m}(\mathbf{X}_* \mid \theta) + \mathbf{K}\left(\mathbf{X}_*, \mathbf{X} \mid \theta\right)\left[\mathbf{K}(\mathbf{X}, \mathbf{X} \mid \theta) + \sigma^2\mathbf{I}\right]^{-1}\left(\mathbf{y} - \mathbf{m}(\mathbf{X} \mid \theta)\right) \tag{4a}$$

$$\mathbf{\Sigma}_*(\mathbf{X}_* \mid \theta) = \mathbb{V}(\mathbf{f}_*)$$
$$= \mathbf{K}\left(\mathbf{X}_*, \mathbf{X}_* \mid \theta\right) - \mathbf{K}\left(\mathbf{X}_*, \mathbf{X} \mid \theta\right)\left[\mathbf{K}(\mathbf{X}, \mathbf{X} \mid \theta) + \sigma^2\mathbf{I}\right]^{-1}\mathbf{K}\left(\mathbf{X}, \mathbf{X}_* \mid \theta\right) \tag{4b}$$

For matrix input $\mathbf{X}_*$, we denote the posterior prediction mean as vector $\mu_*(\mathbf{X}_*)$ and (co)variance as matrix $\Sigma_*(\mathbf{X}_*)$; similarly, for vector input $\mathbf{x}_*$, we denote the outputs as scalars $\mu_*(\mathbf{x}_*)$ and $\sigma_*^2(\mathbf{x}_*)$, respectively.

Most importantly, GP models by construction quantify prediction uncertainty via Eq. (4b). Furthermore, GPs are non-parametric regression models which means the training data $(\mathbf{X}, \mathbf{y})$ are directly embedded into the model. As consequence, a GP model interpolates between the training data and the kernel function describes the prediction uncertainty as a function of distance (for stationary models) from the training data.

GP Training via Hyperparameter Optimization

The GP hyperparameters θ may be chosen based on the modeler's intuition, e.g., σ corresponds to the random error of an experiment. In fact, Eq. (4) can be used to make predictions with any valid values of θ, e.g., $\ell > 0$, $\mathbf{K}$ is positive semi definite, see [15] for details. Often maximum likelihood estimation (MLE) is performed to infer the hyperparameters θ from the data $\mathbf{X}$ and $\mathbf{y}$:

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

In Eq. (5), $p(\mathbf{y} \mid \mathbf{X}, \theta)$ is the GP likelihood function. In MLE, the log of the likelihood function is maximized by searching over θ using derivative-based nonlinear programming algorithms. This is a nonconvex optimization problem that often exhibits many local maxima; multi-start initialization is strongly recommended. Alternatively, a leave-one-out cross-validation objective may be used for hyperparameter tuning. See Chapter 5 of [14] for additional details.

Acquisition Functions Balance Exploitation and Exploration

The main task for the acquisition function (AF) is to find the optimal next experimental conditions $\bar{\mathbf{x}}$ by balancing the trade-off between exploitation and exploration.[12] When the AF favors for exploitation, it suggests $\bar{\mathbf{x}}$ near the current best-known decision $\mathbf{x}^+$ to maximize $f(\bar{\mathbf{x}})$; conversely, when the AF favors exploration, it recommends decisions with high prediction uncertainty. Expected Improvement (EI) is the most popular AF for materials and product design BO, and is defined as the expected value of $\max(f(\bar{\mathbf{x}}) - f(\mathbf{x}^+),\ 0)$. When using a GP surrogate model for $f(\cdot)$, EI is analytically calculated as follows[16]:

$$\mathrm{EI}(\bar{\mathbf{x}}) = \begin{cases} \underbrace{\left(\mu_*(\bar{\mathbf{x}}) - f\left(\mathbf{x}^+\right) - \zeta\right)\Phi(z)}_{\text{exploitation}} + \underbrace{\sigma_*(\bar{\mathbf{x}})\phi(z)}_{\text{exploration}}, & \sigma_*(\bar{\mathbf{x}}) > 0 \\ 0, \quad \text{otherwise} \end{cases} \tag{6a}$$

$$z(\bar{\mathbf{x}}) = \begin{cases} \frac{\mu_*(\bar{\mathbf{x}}) - f\left(\mathbf{x}^+\right) - \zeta}{\sigma_*(\bar{\mathbf{x}})}, & \sigma_*(\bar{\mathbf{x}}) > 0 \\ 0, \quad \text{otherwise} \end{cases} \tag{6b}$$

Here $\Phi(z)$ and $\psi(z)$ are the cumulative and probability density functions for the standard normal distribution, respectively, $f(\mathbf{x}^+)$ is the largest measured value, i.e., $\mathbf{x}^+ = \mathrm{argmax}(f(\mathbf{x}_1), ..., f(\mathbf{x}_n))$, and ζ is an adjustable parameter. For large values of ζ, Eq. (6) favors exploration by placing less importance on the posterior prediction mean $\mu_*(\bar{\mathbf{x}})$ and thus increasing the relative importance of $\sigma_*(\bar{\mathbf{x}})$. Conversely, smaller values of ζ favor exploitation. Figure 1 shows BO applied to a test function using EI with $\zeta = 0$ (no bias towards exploration).

Extension to other surrogate models and acquisition functions

As previously discussed, GP models are non-parametric and include the training data $(\mathbf{X}, \mathbf{y})$. Unfortunately, the matrix inversions in Eq. (4), which are often computed via Cholesky decomposition, are too computationally expensive with more than 10,000 observations without deploying specialized modeling and numerical techniques (see Chapter 8 in [14]). Recent work focuses on adapting BO to deep learning models better suited for large datasets.[17, 18] Other AFs

include portfolio allocation, entropy-based acquisition function, knowledge gradient, and mean objective cost of uncertainty.[16] The general idea of balancing the trade-off between exploitation and exploration is universal across AFs, although the relative performance is problem specific. Many practitioners start with EI and then explore other AFs as they develop BO frameworks for new applications.

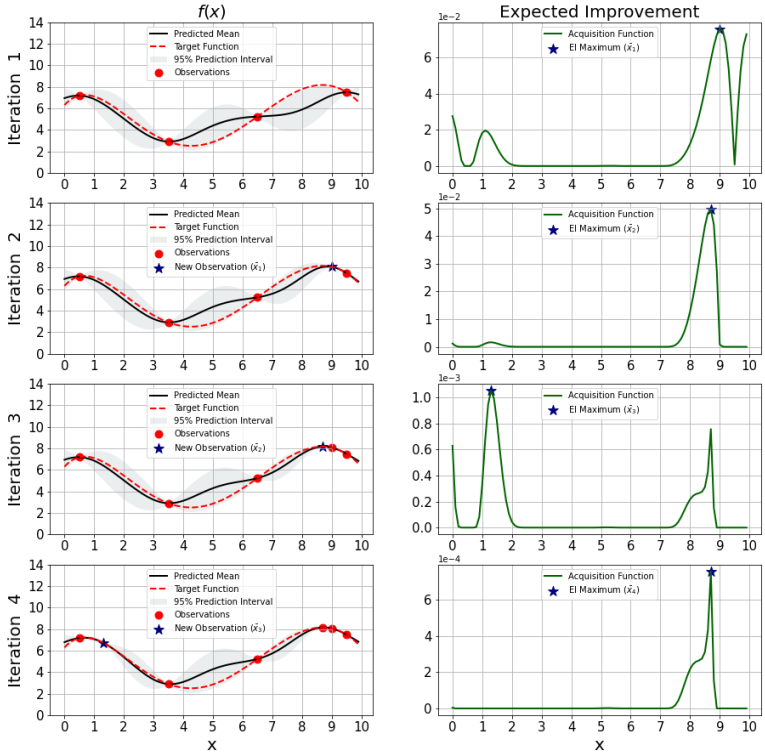

Figure 1: **Bayesian optimization illustrative example.** BO with EI is applied to the test problem $f(x) = 0.0015x^5 - 0.055x^4 + 0.65x^3 - 2.8x^2 + 3.0x + 6.3$ using a Squared Exponential kernel (k_{SE}) with hyperparameter $\theta_{SE} = 1$ and $\sigma = 10^{-3}$. The **left column** shows the GP prediction mean (black line) and 95% prediction intervals (grey regions), the unknown function $f(x)$ (red dashed line), and available training data for each iteration (red dots). The **right column** shows the recommend optimal sample (blue star) which maximizes the EI acquisition function (green line). **Iteration 1**: We randomly select 4 values of x on the interval $0 \leq x \leq 10$, sample the unknown function $f(x)$ with $\mathcal{N}(0, \sigma^2)$ observation error, and construct the GP model with the fixed hyperparameter values. We find $\bar{x}_1 = 9.0$ maximizes EI. **Iteration 2**: We query $f(\bar{x}_1)$ and add the observation error, update the GP model to incorporate these new data, and find $\bar{x}_2 = 8.7$ maximizes the EI. (For simplicity, we keep the hyperparameters fixed.) **Iteration 3**: After sampling $f(\bar{x}_2)$ and updating the GP model, we find $\bar{x}_3 = 1.3$. **Iteration 4**: We sample $f(\bar{x}_3)$, retrain the GP model, and find $\bar{x}_4 = 8.7$ **Summary**: Through this example, we show BO can efficiently sample the unknown function $f(x)$ near its two local maxima at $x = 1.3$ and $x = 8.7$ by balancing exploitation and exploration. By comparing iterations 1 to 4, we see the prediction intervals (grey regions) shrink as more observations are added to the GP surrogate model.

How can BO accelerate materials and product design?

Traditional materials and product design harmonize empirical data, scientific intuition, and computational methods that describe the behavior of matter at the atomistic and (macro)molecular [19] to iteratively discover materials that enable new products. Although successful, these methods rely on expensive and slow computational and physical experiments. Moreover, it is difficult for human-driven scientific intuition to navigate trade-offs in high-dimensional design spaces. BO-driven inverse design frameworks systematically guide prevailing Edisonian workflows with artificial intelligence to accelerate molecular, materials, and product innovations.[20] The previously described 6-step general BO workflow is easily adapted to the inverse design of new molecules, materials, or products with one or more tailored physical properties by optimizing physical or computational experiments.

Molecular design and discovery

Artificial Intelligence (AI) offers new paradigms to systematically design and discover (macro)molecules with specific properties and functionality.[10] For example, generative machine learning models including generative adversarial networks (GAN), reinforcement learning (RL), recurrent neural networks (RNN), and variational autoencoders (VAE) are commonly used to propose new molecular structures.[10] For brevity, we focus this review on variational autoencoder-based Bayesian optimization (VAE-BO), which was first proposed by Gómez-Bombarelli et al. [21]. VAE models were first proposed by Kingma et al.[22] and have two main components. The VAE encoder is an RNN or a similar deep learning model that converts string-like molecular representations (e.g., SMILES strings) into a low to medium-dimensional vector in continuous latent space. Similarly, the VAE decoder converts a vector in the latent space into a string-like molecular structure. BO is then performed in the continuous latent space. Unfortunately, many points in latent space do not map to valid molecular structures. To overcome this challenge, Griffiths et al. [23] proposed

a constrained BO approach that uses an additional surrogate model to predict the probability that each point in latent space corresponds to a valid molecule.

Table 1 highlights diverse applications of VAE and BO for materials design including proteins[4], ligands (binding sites)[24], drug discovery[21, 23, 25], and light-absorbing organic molecules[26]. Moreover, Table 1 highlights how BO has been integrated with atomistic or (macro)molecular physics-based models to further accelerate materials design. For example, Tamura et al.[27] used BO to estimate spin-spin interactions using magnetization curve synthesized by experimental data, successfully selecting relevant terms of interactions from redundant candidates with high accuracy. Ju et al.[28] integrate atomistic Green's function and BO to maximize and minimize interfacial thermal conductance; they ultimately identify a non-intuitive optimal structure with 50% improvement. Yan et al.[29] integrate high throughput calculations with BO to discover atomistic SiGe alloy configurations with extremely low thermal conductivity. Sestito et al. [30] use multiobjective Bayesian optimization to calibrate molecular dynamics force fields for polycaprolactone.

Smart and additive manufacturing

Under the emerging smart manufacturing paradigm, intelligent systems and robotics are augmenting the synthesis and fabrication of new materials.[31] Additive manufacturing refers to layer-upon-layer fabrication processes controlled by an intelligent system such as a computer-aided design (CAD) model [32] and enables the creation of new multi-functional materials that cannot be manufactured via conventional methods.[33] However, optimizing smart and additive manufacturing processes is challenging because these systems are often too complex to describe with physics-based computational models. This is especially true for human-in-the-loop cyber-physical systems. BO enables automated learning and self-optimization to address this challenge.

BO improves smart and additive manufacturing systems, often an order of magnitude fewer experiments (less data) than trial-and-error Edisonian search.[13] Many of these applications integrate BO with new additive manufacturing and

Ref.	Application	Finding	Method
[21]	Drug-like molecular design	Seminal paper demonstrating VAE autoencoder with low reconstruction error rates (i.e., generates few invalid molecules) and proof-of-concept BO in molecular latent space.	VAE-BO
[23]	Drug-like molecular design	Seminal paper demonstrating constrained BO to reduce the number of invalid generated molecules in VAE model.	VAE-CBO
[25]	Drug-like molecular design	A concrete example to illustrate the effectiveness of BO in complex multi-optimal drug-design problems comparing with random search and greedy search.	BO
[4]	Protein sequence design	Proposes VAE framework to capture complex protein sequence-function relationships and applies BO to identify mutations that optimize specific protein functions.	VAE-BO
[26]	Organic light-absorbing molecular design	Demonstrates VAE-BO framework trained on simple molecular structures can generate complex molecular structures.	VAE-BO
[24]	Ligand design	VAE-BO identifies sub-regions of the ligand design space with improved binding sites.	VAE-BO
[27]	Magnetic material modeling	Leverages BO to calibrate spin-spin interactions in molecular models using experimental data.	BO
[28]	SiGe composite design	Use BO to identify a non-intuitive optimal structure for Si-Si and Si-Ge composite which delivers 50% improved thermal conductive compared to the state-of-the-art.	BO
[29]	SiGe composite design	BO identifies new layers structures to minimize the thermal conductivity of SiGe alloys.	BO

Table 1: Recent applications of BO in molecular modeling and design.

high throughput characterization capabilities to optimize the formulation of chemical products and materials. For example, Nakano et al.[5] optimized Li, Ca, Y, and Zr composition in solid electrolytes to maximize Li-ion conductivity. They show BO with only 40 observations outperforms high throughput trial-and-error search with 169 observations. Lookman et al.[34] applied BO to minimize the thermal dissipation of shape memory alloys by optimizing the composition (Ni, Cu, Pd, Fe, Ti). Despite the vast search space with 800,000

possible alloy compositions, they discovered a novel alloy with 42% improvement in only 9 iterations (with 4 alloy samples per iteration). Gongora et al.[35] developed the Bayesian experimental autonomous researcher (BEAR) and also demonstrate a 60-fold reduction in the number of experiments to optimize additive manufacturing structures versus grid search. In addition to the aforementioned literature, Table 2 summarizes the wide application of BO to optimizing materials[36, 5], alloys[37, 34], and additive manufacturing processes[35, 38].

Ref.	**Application**	**Finding**	**Decision Variables**
[36]	High performance concrete formulation	Demonstrates inverse design by optimizing 7 variables and shows the predictions agree with physicochemical measurements.	Quantity of cement, water, fly ash, etc.
[5]	Solid electrolyte design	BO only requires 40 experiments to maximize Li-ion conductivity compared to 169 with trial-and-error search.	Composition of Li, Ca, Y, Zr
[37]	Mesoporous alloy design	BO only required 47 experiments to optimize catalytic activity.	Composition of $AcCl_4$, $PtCl_4$, $PdCl_4$
[34]	NiTi alloy design	BO identifies novel alloy by efficiently searching over 800,000 possible compositions using only 36 experiments.	Ratio of Ni, Cu, Pd, Fe
[35]	Additive manufacturing	5 out of 100 mechanical structures designed by BO outperformed 1,800 designs generated via grid search.	Number of hollow columns, outer radius, thickness, twist angle
[38]	Multi-material additive manufacturing for composite solids	VAE-BO successfully optimizes macroscopic elastic moduli in lattice structured composite material.	Representative volume element (modeled with VAE)

Table 2: Recent applications of BO in smart and additive manufacturing.

What are the current challenges and future opportunities?

Leveraging complex data

Experiments are often expensive and involve multiple data sources with complex uncertainty structures; multi-fidelity BO is an emerging technique to systematically fuse data from multiple physical or computational experiments.[16] The general idea is to use inexpensive but less accurate models for initial exploration and transition to higher-fidelity experiments to refine the search. This is done by leveraging the correlation between low and high-fidelity experiments. For example, Herbol et al.[39] recently proposed a BO framework to fuse data from multiple experiments using the Pearson correlation coefficient. Another challenge is heteroscedastic uncertainty structures. Recall we previously assumed observations each observation is corrupted by $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ measurement error. We assumed this error was homoscedastic, i.e., it has constant (but perhaps unknown) variance σ^2. Griffiths et al.[40] discuss the benefits of using heteroscedastic uncertainty for BO.

BO with discrete decisions

Many materials and manufacturing optimization problems involve both discrete and continuous decisions. However, a vast majority of BO frameworks only support continuous decision variables. Maximizing acquisition functions in a nonconvex bounded continuous optimization problem is already challenging due to the presence of local optima. Extending BO to include discrete decisions yields a nonconvex mixed-integer nonlinear optimization problem which is especially difficult to solve numerically. As a workaround, discrete decisions are relaxed to continuous variables, the BO optimization problem is numerically solved, and rounding is applied.[41] Although simple to implement, it is well known that rounding can yield sub-optimal results. Alternatively, Zhang et al.[42] proposed a latent-variable Gaussian process (LVGP) to map discrete decisions into numerical latent space in GP; Ru et al.[43] designed a novel kernel structure that incorporates both decisions in GP. They showed improved prediction accuracy with fewer iterations. In this regard, VAE models transform

discrete decisions for molecular design into a continuous latent space; unfortunately, these continuous latent spaces, especially if high dimensional, often contain many local optima[44]. To address this, Eriksson et al.[45] propose to replace the global surrogate model with multiple local surrogate models to more efficiently discover global optimal solutions.

Batch experiment optimization

Often data is collected in parallel through high throughput screening (e.g., deposition of films with continuous concentration gradients) or high-performance computing. This requires BO to recommend the next experiments in batches.[46] One common approach considers a batch of GP posteriors. For example, Joy et al.[47] train an ensemble of GP models with different kernels and hyperparameters and maximize the acquisition function for each GP model separately to assemble BO batches. Similarly, Snoek et al.[48] and Ginsbourger et al.[49] consider a batch of posteriors as a proxy to fit the acquisition function in multi-dimensions that delivered BO batches. Alternatively, González et al.[50] assemble batches from multiple local maxima of the acquisition function.

Open-source software platforms

Special care is required when choosing a BO software platform for molecular design, material discovery, or manufacturing optimization. For example, COMBO[51], one of the first open-source BO platforms, has not been actively developed for almost three years. Similarly, pyGPGO[52] was last updated about two years ago. The three packages most actively developed at the time of writing are ChemOS[53], BoTorch[54], and pyOpt[55]. BO is a rapidly evolving methodology with new surrogate models and acquisition functions proposed each year, and unfortunately, no single BO software platform implements every innovation. We recommend practitioners compare the features, documentation, and tutorials for a handful of BO software platforms before starting each new project. We also recommend new BO users consult the excellent hyperparameter tuning study by Cowen-Rivers et al.[56]

Outlook

In this review, we highlight the recent success of BO applied to the diverse application in products and material design including additive manufacturing. Many of the examples demonstrate how BO requires orders of magnitude few experiments (physical or computational) than Edisonian search. Based on these results, we anticipate BO will continue to grow in popularity, especially for product and materials applications that are challenging to model from first principles. Nevertheless, there is a need (and opportunity) to systematically benchmark BO against other emerging machine learning architectures for molecular design [57] and similar applications.

Acknowledgements

This work was partially supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Advanced Manufacturing Office Award Number DE-EE0009103.

Conflict of interest statement

The authors declare no conflict of interest

References

[1] L. Zhang, H. Mao, Q. Liu, R. Gani, Chemical product design–recent advances and perspectives, Current Opinion in Chemical Engineering 27 (2020) 22–34,

⋆ ⋆ This paper provides an excellent review of chemical-product design challenges for physical and data-based methods.

[2] F. Mushtaq, X. Zhang, K. Y. Fung, K. M. Ng, Computational design of structured chemical products, Frontiers of Chemical Science and Engineering (2021) 1–17

⋆ This paper reviews computation design of micro- and nanostructured materials and products.

[3] X. Zhang, T. Zhou, L. Zhang, K. Y. Fung, K. M. Ng, Food product design: a hybrid machine learning and mechanistic modeling approach, Industrial & Engineering Chemistry Research 58 (36) (2019) 16743–16752.

[4] S. Sinai, E. Kelsic, G. M. Church, M. A. Nowak, Variational auto-encoding of protein sequences, arXiv preprint arXiv:1712.03346 (2017).

[5] K. Nakano, Y. Noda, N. Tanibata, H. Takeda, M. Nakayama, R. Kobayashi, I. Takeuchi, Exhaustive and informatics-aided search for fast Li-ion conductor with NASICON-type structure using material simulation and Bayesian optimization, APL Materials 8 (4) (2020) 041112.

[6] C. L. Hanselman, D. R. Alfonso, J. W. Lekse, C. Matranga, D. C. Miller, C. E. Gounaris, et al., A framework for optimizing oxygen vacancy formation in doped perovskites, Computers & Chemical Engineering 126 (2019) 168–177.

[7] D. C. Elton, Z. Boukouvalas, M. D. Fuge, P. W. Chung, Deep learning for molecular design—a review of the state of the art, Molecular Systems Design & Engineering 4 (4) (2019) 828–849,
⋆ ⋆ This paper provides an excellent introduction to deep learning-based molecular design.

[8] E. A. Eugene, W. A. Phillip, A. W. Dowling, Data science-enabled molecular-to-systems engineering for sustainable water treatment, Current Opinion in Chemical Engineering 26 (2019) 122–130.

[9] N. D. Austin, N. V. Sahinidis, D. W. Trahan, Computer-aided molecular design: An introduction and review of tools, applications, and solution techniques, Chemical Engineering Research and Design 116 (2016) 2–26.

[10] K. T. Schütt, S. Chmiela, O. A. von Lilienfeld, A. Tkatchenko, K. Tsuda, K.-R. Müller, Schütt, Scalone, Machine Learning Meets Quantum Physics,

Springer, 2020,
⋆ This textbook offers a thorough introduction to machine learning for molecular design and highlights both physical and data-driven perspectives.

[11] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, N. De Freitas, Taking the human out of the loop: A review of Bayesian optimization, Proceedings of the IEEE 104 (1) (2015) 148–175.

[12] P. I. Frazier, Bayesian optimization, in: Recent Advances in Optimization and Modeling of Contemporary Problems, INFORMS, 2018, pp. 255–278.

[13] T. Lookman, P. V. Balachandran, D. Xue, R. Yuan, Active learning in materials science with emphasis on adaptive sampling using uncertainties for targeted design, npj Computational Materials 5 (1) (2019) 1–17.

[14] C. E. Rasmussen, C. I. Williams, Gaussian Processes for Machine Learning, MIT Press, 2006.

[15] R. B. Gramacy, Surrogates: Gaussian process modeling, design, and optimization for the applied sciences, CRC Press, 2020,
⋆ This textbook offers an excellent introduction to Gaussian Process surrogates and modern methods for adaptive design of experiments.

[16] F. Archetti, A. Candelieri, Bayesian optimization and data science, Springer, 2019.

[17] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, A. Walsh, Machine learning for molecular and materials science, Nature 559 (7715) (2018) 547–555.

[18] T. Wang, M. Shao, R. Guo, F. Tao, G. Zhang, H. Snoussi, X. Tang, Surrogate model via artificial intelligence method for accelerating screening materials and performance prediction, Advanced Functional Materials 31 (8) (2021) 2006245.

[19] E. J. Maginn, From discovery to data: What must happen for molecular simulation to become a mainstream chemical engineering tool, AIChE Journal 55 (6) (2009) 1304–1310.

[20] A. Zunger, Inverse design in search of materials with target functionalities, Nature Reviews Chemistry 2 (4) (2018) 1–16.

[21] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, A. Aspuru-Guzik, Automatic chemical design using a data-driven continuous representation of molecules, ACS Central Science 4 (2) (2018) 268–276.

[22] D. P. Kingma, M. Welling, Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114 (2013).

[23] R.-R. Griffiths, J. M. Hernández-Lobato, Constrained Bayesian optimization for automatic chemical design using variational autoencoders, Chemical Science 11 (2) (2020) 577–586.

[24] V. Mallet, C. G. Oliver, N. Moitessier, J. Waldispuhl, Leveraging binding-site structure for drug discovery with point-cloud methods, arXiv preprint arXiv:1905.12033 (2019).

[25] E. O. Pyzer-Knapp, Bayesian optimization for accelerated drug discovery, IBM Journal of Research and Development 62 (6) (2018) 2–1.

[26] K. Kim, S. Kang, J. Yoo, Y. Kwon, Y. Nam, D. Lee, I. Kim, Y.-S. Choi, Y. Jung, S. Kim, et al., Deep-learning-based inverse design model for intelligent discovery of organic molecules, npj Computational Materials 4 (1) (2018) 1–7.

[27] R. Tamura, K. Hukushima, Method for estimating spin-spin interactions from magnetization curves, Physical Review B 95 (6) (2017) 064407.

[28] S. Ju, T. Shiga, L. Feng, Z. Hou, K. Tsuda, J. Shiomi, Designing nanostructures for phonon transport via Bayesian optimization, Physical Review X 7 (2) (2017) 021024.

[29] J. Yan, H. Wei, H. Xie, X. Gu, H. Bao, Seeking for low thermal conductivity atomic configurations in SiGe alloys with Bayesian optimization, ES Energy & Environment 8 (2) (2020) 56–64.

[30] J. M. Sestito, M. L. Thatcher, L. Shu, T. A. Harris, Y. Wang, Coarse-grained force field calibration based on multiobjective Bayesian optimization to simulate water diffusion in poly-ε-caprolactone, J. Phys. Chem. A 124 (24) (2020) 5042–5052.

[31] C. Shang, F. You, Data analytics and machine learning for smart process manufacturing: recent advances and perspectives in the big data era, Engineering 5 (6) (2019) 1010–1016.

[32] L. Meng, B. McWilliams, W. Jarosinski, H.-Y. Park, Y.-G. Jung, J. Lee, J. Zhang, Machine learning in additive manufacturing: A review, Jom 72 (6) (2020) 2363–2377,
⋆ This paper highlights recent advances in machine learning for additive manufacturing.

[33] Z. Jin, Z. Zhang, K. Demir, G. X. Gu, Machine learning for advanced additive manufacturing, Matter 3 (5) (2020) 1541–1556.

[34] T. Lookman, P. V. Balachandran, D. Xue, J. Hogden, J. Theiler, Statistical inference and adaptive design for materials discovery, Current Opinion in Solid State and Materials Science 21 (3) (2017) 121–128.

[35] A. E. Gongora, B. Xu, W. Perry, C. Okoye, P. Riley, K. G. Reyes, E. F. Morgan, K. A. Brown, A Bayesian experimental autonomous researcher for mechanical design, Science Advances 6 (15) (2020) eaaz1708.

[36] X. Ke, Y. Duan, A Bayesian machine learning approach for inverse prediction of high-performance concrete ingredients with targeted performance, Construction and Building Materials 270 (2021) 121424.

[37] A. S. Nugraha, G. Lambard, J. Na, M. S. A. Hossain, T. Asahi, W. Chaikittisilp, Y. Yamauchi, Mesoporous trimetallic ptpdau alloy films toward enhanced electrocatalytic activity in methanol oxidation: unexpected chemical compositions discovered by Bayesian optimization, Journal of Materials Chemistry A 8 (27) (2020) 13532–13540.

[38] T. Xue, T. J. Wallin, Y. Menguc, S. Adriaenssens, M. Chiaramonte, Machine learning generative models for automatic design of multi-material 3D printed composite solids, Extreme Mechanics Letters 41 (2020) 100992.

[39] H. C. Herbol, M. Poloczek, P. Clancy, Cost-effective materials discovery: Bayesian optimization across multiple information sources, Materials Horizons 7 (8) (2020) 2113–2123.

[40] R.-R. Griffiths, A. A. Aldrick, M. Garcia-Ortegon, V. R. Lalchand, A. A. Lee, Achieving robustness to aleatoric uncertainty with heteroscedastic Bayesian optimisation, arXiv preprint arXiv:1910.07779 (2019).

[41] A. Tran, M. Tran, Y. Wang, Constrained mixed-integer gaussian mixture Bayesian optimization and its applications in designing fractal and auxetic metamaterials, Structural and Multidisciplinary Optimization 59 (6) (2019) 2131–2154.

[42] Y. Zhang, D. W. Apley, W. Chen, Bayesian optimization for materials design with mixed quantitative and qualitative variables, Scientific Reports 10 (1) (2020) 1–13.

[43] B. Ru, A. Alvi, V. Nguyen, M. A. Osborne, S. Roberts, Bayesian optimisation over multiple continuous and categorical inputs, in: International Conference on Machine Learning, PMLR, 2020, pp. 8276–8285.

[44] A. Grosnit, R. Tutunov, A. M. Maraval, R.-R. Griffiths, A. I. Cowen-Rivers, L. Yang, L. Zhu, W. Lyu, Z. Chen, J. Wang, et al., High-dimensional Bayesian optimisation with variational autoencoders and deep metric learning, arXiv preprint arXiv:2106.03609 (2021).

[45] D. Eriksson, M. Pearce, J. Gardner, R. D. Turner, M. Poloczek, Scalable global optimization via local Bayesian optimization, Advances in Neural Information Processing Systems 32 (2019) 5496–5507.

[46] R. Couperthwaite, A. Molkeri, D. Khatamsaz, A. Srivastava, D. Allaire, R. Arròyave, Materials design through batch Bayesian optimization with multisource information fusion, JOM 72 (12) (2020) 4431–4443.

[47] T. T. Joy, S. Rana, S. Gupta, S. Venkatesh, Batch Bayesian optimization using multi-scale search, Knowledge-Based Systems 187 (2020) 104818.

[48] J. Snoek, H. Larochelle, R. P. Adams, Practical Bayesian optimization of machine learning algorithms, Advances in Neural Information Processing Systems 25 (2012).

[49] D. Ginsbourger, R. Le Riche, L. Carraro, Kriging is well-suited to parallelize optimization, in: Computational Intelligence in Expensive Optimization Problems, Springer, 2010, pp. 131–162.

[50] J. González, Z. Dai, P. Hennig, N. Lawrence, Batch Bayesian optimization via local penalization, in: Artificial Intelligence and Statistics, PMLR, 2016, pp. 648–657.

[51] T. Ueno, T. D. Rhone, Z. Hou, T. Mizoguchi, K. Tsuda, COMBO: an efficient Bayesian optimization library for materials science, Materials Discovery 4 (2016) 18–21.

[52] J. Jiménez, J. Ginebra, pyGPGO: Bayesian optimization for python, Journal of Open Source Software 2 (19) (2017) 431.

[53] L. M. Roch, F. Häse, C. Kreisbeck, T. Tamayo-Mendoza, L. P. Yunker, J. E. Hein, A. Aspuru-Guzik, ChemOS: An orchestration software to democratize autonomous discovery, PLoS One 15 (4) (2020) e0229862.

[54] M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson, E. Bakshy, BoTorch: Programmable Bayesian optimization in PyTorch, arXiv preprint arXiv:1910.06403 (2019).

[55] R. E. Perez, P. W. Jansen, J. R. Martins, pyOpt: a python-based object-oriented framework for nonlinear constrained optimization, Structural and Multidisciplinary Optimization 45 (1) (2012) 101–118.

[56] A. I. Cowen-Rivers, W. Lyu, R. Tutunov, Z. Wang, A. Grosnit, R. R. Griffiths, H. Jianye, J. Wang, H. B. Ammar, An empirical study of assumptions in Bayesian optimisation, arXiv preprint arXiv:2012.03826 (2020).

[57] N. Brown, M. Fiscato, M. H. Segler, A. C. Vaucher, Guacamol: benchmarking models for de novo molecular design, Journal of Chemical Information and Modeling 59 (3) (2019) 1096–1108.